Calhoun: The NPS Institutional Archive
DSpace Repository

Theses and Dissertations 1. Thesis and Dissertation Collection, all items

1978-12

Reliability control model for stored items requiring rework.

Ferreira, Paulo Antonio

http://hdl.handle.net/10945/18536

Copyright is reserved by the copyright owner

Downloaded from NPS Archive: Calhoun

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93940

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

RELIABILITY CONTROL MODEL FOR STORED ITEMS REQUIRING REWORK

by

Paulo Antonio Ferreira

December 1978

Thesis Advisor: G.F. Lindsay

Approved for public release; distribution unlimited.

T187416

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
Reliability Control Model for Stored Items Requiring Rework

5. TYPE OF REPORT & PERIOD COVERED
Master's Thesis;
December 1978

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Paulo Antonio Ferreira

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Naval Postgraduate School
Monterey, California 93940

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Naval Postgraduate School
Monterey, California 93940

12. REPORT DATE
December 1978

13. NUMBER OF PAGES
116

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Reliability Control Model
Inventory Management
Control Theory
Kalman Filter
Optimal Control
Dynamic Programming

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An application of control theory to an administrative problem is given for the case of a system of stored items which are periodically reworked to improve their reliability. Expressions are developed for the final value of the reliability when the system is stable and the limits of stability are found. A Kalman filter is used in the control model to obtain an estimation of the item reliability when there are random errors in the measurement

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601 |

ABSTRACT (Cont'd)

and in the rework process. An extension is done for more than one dimension for systems composed of subsystems in series, parallel or a combination of both. A procedure for an optimal sequence of levels of rework is found in the sense of optimizing a linear combination of several performance measures. Numerical examples are presented to demonstrate the use of the several expressions.

Approved for public release; distribution unlimited.

Reliability Control Model for Stored Items Requiring Rework

by

Paulo Antonio Ferreira
Lieutenant Commander, Brazilian Navy
B.S., Naval Postgraduate School, 1978

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
December 1978

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93940

ABSTRACT

An application of control theory to an administrative problem is given for the case of a system of stored items which are periodically reworked to improve their reliability. Expressions are developed for the final value of the reliability when the system is stable and the limits of stability are found. A Kalman filter is used in the control model to obtain an estimation of the item reliability when there are random errors in the measurement and in the rework process. An extension is done for more than one dimension for systems composed of subsystems in series, parallel or a combination of both. A procedure for an optimal sequence of levels of rework is found in the sense of optimizing a linear combination of several performance measures. Numerical examples are presented to demonstrate the use of the several expressions.

TABLE OF CONTENTS

I. INTRODUCTION -- 6

II. DEVELOPMENT OF THE REWORK MODEL ---------------- 9

- A. THE EFFECTIVENESS OF REWORK ---------------- 9
- B. THE REWORK MODEL ----------------------------- 10
- C. THE SOLUTION TO THE REWORK MODEL ------------ 15
- D. THE CONTROL VARIABLE ------------------------ 20

III. DEVELOPMENT OF THE CONTROL MODEL ---------------- 22

- A. THE CONTROL THEORY APPROACH ----------------- 23
- B. THE RELIABILITY CONTROL MODEL --------------- 31

IV. APPLICATIONS OF THE CONTROL THEORY MODEL -------- 39

- A. SYSTEM STABILITY ---------------------------- 39
- B. THE FINAL VALUE IN STABLE SYSTEMS ----------- 41
- C. ERROR COMPENSATION USING A KALMAN FILTER ---- 43
- D. EXTENSIONS TO MORE THAN ONE DIMENSION ------- 53

V. OPTIMIZATION -- 58

- A. THE PERFORMANCE MEASURE ---------------------- 58
- B. OPTIMIZATION USING DYNAMIC PROGRAMMING ------ 61

VI. CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER STUDY -- 73

APPENDIX A: Z TRANSFORM ------------------------------ 76

APPENDIX B: COMPUTER PROGRAM DISCUSSION -------------- 87

COMPUTER OUTPUT --------------------------------------- 89

COMPUTER PROGRAM -------------------------------------- 109

BIBLIOGRAPHY -- 115

INITIAL DISTRIBUTION LIST ----------------------------- 116

I. INTRODUCTION

In an earlier work Bohannan [2] developed a mathematical model for a system of stored items which are periodically reworked to improve their reliability. That model yields an expression, for the reliability of an item following rework, in the form of a series. More recently, Bishop [1] applied the discrete control theory to systems encountered in economics and operations research rather than the more typical applications in electrical systems.

In this thesis the mathematical model for the system of stored items which are periodically reworked is developed in terms of control theory, and the solution obtained for item reliability has a closed form that is more suitable for further studies in this area. Important characteristics such as system stability and the steady state value of the reliability are directly derived from the control model rather than obtained by inspection of simulated values.

The reliability of the stored items will depend upon the reliability at the time of acquisition together with the storage environment and length of time the items are stored. When periodic rework of items is done, other variables that can affect the reliability of the item are the effectiveness of the rework, the rate of rework, the rate of acquisition of new items and the rate and policy of expenditure of items for use, obsolescence, or even training purposes.

Such a system might include a stock of ordnance which is acquired, stored, and periodically reworked, but not expended except for war time use, or alternatively expended in training with replacement by new items.

On the other hand, items such as big missiles are not likely to be expended, and thus expenditure and replacement cannot be considered as a control variable for the reliability. Thus, unless due to other factors, the expenditure of such items would be avoided in favor of rework of the items.

In this thesis we will study only the case where there is no expenditure and replacement, leaving reliability over time to be maintained by rework. We will also consider the case where we have several levels of rework and want to find the optimal sequence of levels over successive reworks in order to optimize a given performance measure.

A general rework model is developed in Chapter II which relates the reliability of an item following rework to its reliability following the last rework. An equivalent model is also developed relating the reliabilities of an item immediately before consecutive reworks. These models may be solved for any rework cycle given the initial reliability at acquisition.

In Chapter III the concepts of control theory are reviewed and applications to control of the reliability of items in inventory are suggested and structured.

When we have random errors in our measure of the item reliability or when the rework process introduces randomness in the item reliability, then we have to make our judgments based on estimated values. In Chapter IV, a Kalman Filter is used with our control model to accomplish this estimation. An examination of the final value and the stability of our system is followed by an extension to more than one dimension for the case where we have systems composed of several subsystems in series, parallel or a combination of both.

In Chapter V we shall extend our work to the case where we can decide among several levels of rework. Using dynamic programming, we will show how to derive the optimal rework policy for this case in accordance with several performance measures, namely, to obtain a desired reliability, to minimize the costs of several reworks, to minimize the time to achieve a desired reliability, or to satisfy a combination of all these performance criteria. Conclusions and recommendations for further study are offered in Chapter VI.

II. DEVELOPMENT OF THE REWORK MODEL

In this chapter the general rework model is developed which will relate either the reliability of an item following a rework or the reliability of an item immediately before a rework to the same reliability for the last rework and the effectiveness of the rework process. A solution is found for these two reliabilities at any instant given the initial or acquisition reliability of the item.

A. THE EFFECTIVENESS OF REWORK

One form of rework mechanism would raise an item's reliability to a certain level which is independent of the item's reliability prior to rework, as might be the case when components or parts are replaced. Another way, developed by Bohannan [2] and adopted in this thesis, is for the rework mechanism to achieve an increase in reliability which is proportional to both the item unreliability before rework and the effectiveness of the rework mechanism. This type of rework mechanism might exist where major assemblies or subassemblies are tested and repaired rather than replaced. Denoting the item reliability immediately before the rework by R, the reliability following the rework by R_s, we can define the effectiveness of the rework process α as

$$\alpha \triangleq \frac{R_s - R}{1 - R} , \qquad 0 \leq \alpha \leq 1 \quad . \qquad (1)$$

The interpretation of the rework effectiveness α is that the greater its value, the more effective the rework process or in other words the item unreliability, 1-R, will be reduced by an amount proportional to α since from Equation (1) we can write

$$1 - R_s = (1-R) - \alpha(1-R).$$

Under these circumstances, it may be possible for such a rework mechanism to improve the item reliability to where it is "better than new", "as good as new" or "not as good as new". Because of deterioration of reliability during storage, we note that in this last case, the reliability might continue to decrease until the items need to be replaced, rather than reworked.

B. THE REWORK MODEL

From the definition of effectiveness of the rework process , given by expression (1), the reliability of an item following the rework, R_s, is

$$R_s = R + \alpha(1-R),$$

or

$$R_s = (1-\alpha)R + \alpha . \qquad (2)$$

Let R(t) represent the reliability of an item of age t when there is no rework. We assume that all failures are random, suggesting (i) that there are no early failures or that some form of "burn in" has been used to eliminate early age failures, and (ii) that because the items are in storage and not in an operating environment, either there are no wearout failures or the time to occurrence of wearout is much longer than the projected time until the next scheduled rework. Thus assuming that the reliability R(t) is an exponential function,

$$R(t) = e^{-(a+bt)} , \qquad (3)$$

for $t \geq 0$. Since the initial reliability R_o, for $t = 0$, is

$$R_o = R(0) = e^{-a} ,$$

we have

$$R(t) = R_o e^{-bt} \qquad (4)$$

for $t \geq 0$ and $b > 0$. The value of the parameter b depends on the nature of the item stored and on the storage environment.

If the system contains N items, which are reworked at a constant rate ρ, then the period T to "turn over" the

inventory is

$$T = \frac{N}{\rho} .$$

We begin by looking at new items that are going to be reworked for the first time. The duration of the rework process is T/N and the "age" of a new item selected for rework under a First In, First Out (FIFO) policy will thus be $T - \frac{T}{N}$, and its reliability R immediately prior to its first rework will be

$$R(T - \frac{T}{N}) = R_o \, e^{-b(T - \frac{T}{N})} \tag{5}$$

If the item is going to be periodically reworked, we can rewrite the above equation for the reliability just prior to the (k+1)st rework. This will occur at time $t = (k+1)T - \frac{T}{N}$, and we have

$$R(kT + T - \frac{T}{N}) = R_s(kT) \, e^{-b(T - \frac{T}{N})}$$

When k = 0 in this equation, we get

$$R(T - \frac{T}{N}) = R_s(0) \, e^{-b(T - \frac{T}{N})}$$

and from Equation (5) we have then

$$R_o = R_s(0) ,$$

or in other words the initial or acquisition reliability may be considered as the "reliability following the 0^{th} rework", or as the initial value for the reliability following the rework, $R_s(k)$. Evidently there is no rework at the time of acquisition and this equivalence is used only for purpose of coherence. The reliability just prior to the first rework is then only defined for values of k equal to or greater than one.

The process is presented schematically in Figure 1. Assuming that N or the rework rate are relatively large, $T - \frac{T}{N} \approx T$ and

$$R(kT+T) = R_s(kT)\ e^{-bT} .$$

For simplicity we will denote R(kT) by R(k), and the reliability immediately before the (k+1)st rework, R(k+1), can then be rewritten as a function of the reliability following the kth rework $R_s(k)$:

$$R(k+1) = R_s(k)\ e^{-bT} \qquad (6)$$

This expresses the reliability of an item immediately before the (k+1)th rework recursively, as an exponential function with initial value $R_s(kT)$ which is the reliability of the item at the beginning of the kth period.

Rewriting expression (2) for the reliability following a rework with this notation, we have

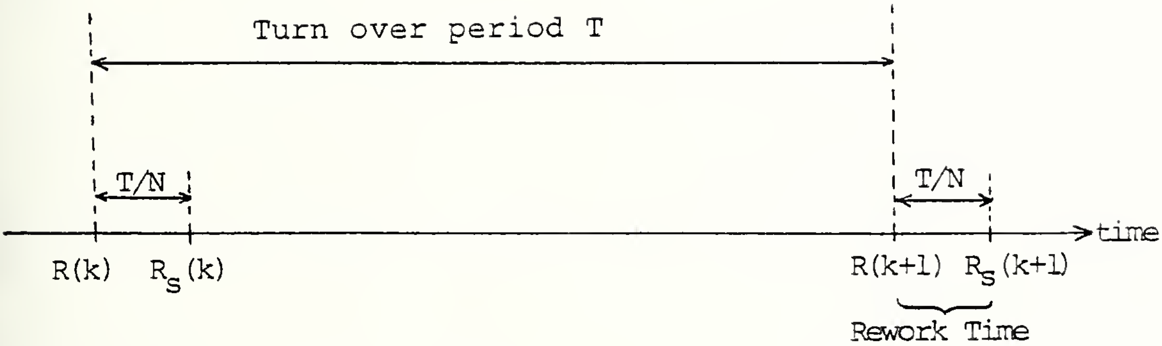

FIGURE 1. Reliabilities during the kth period

$$R_s(k) = (1-\alpha)\ R(k) + \alpha\ . \tag{7}$$

Combining the above equations, we have the expression for the reliability after the (k+1)th rework in terms of the reliability after the kth rework,

$$R_s(k+1) = (1-\alpha)R_s(k)e^{-bT} + \alpha, \quad k = 0,1,2,\ \ldots \tag{8}$$

We may also write the reliability immediately before the (k+1)th rework in terms of the reliability just prior to the kth rework,

$$R(k+1) = R(k)e^{-bT} + \alpha(1-R(k))e^{-bT}, \quad k = 1,2,\ldots \tag{9}$$

When using this equation we have to consider as initial condition the reliability immediately before the first rework, $R(1) = R_o\ e^{-bT}$.

C. THE SOLUTION TO THE REWORK MODEL

Equations (8) and (9) are first order difference equations, and we can solve them by using the z transform technique, described in Appendix A, with the advantage that we will obtain a closed form solution instead of a solution in the form of summation.

In Appendix A we show that taking the z transform of both sides of equation (8), we get

$$zR_s(z) - zR_o = (1-\alpha)\, R_s(z)\, e^{-bT} + \frac{\alpha z}{z-1}, \qquad (10)$$

where $R_s(z)$ means the z transform of $R_s(k)$.

Rearranged, this gives

$$(z - (1-\alpha)e^{-bT})R_s(z) = zR_o + \frac{\alpha z}{z-1},$$

or

$$R_s(z) = \frac{z}{z-(1-\alpha)e^{-bT}}\, R_o + \frac{\alpha z}{(z-(1-\alpha)\ {}^{-bT})(z-1)},$$

which when expanded in partial fractions yields

$$R_s(z) = \frac{z}{z-(1-\alpha)e^{-bT}}\, R_o + \frac{z}{z-1}\,\frac{\alpha}{1-(1-\alpha)e^{-bT}} + \frac{z}{[z-(1-\alpha)e^{-bT}]} \cdot \frac{\alpha}{[(1-\alpha)e^{-bT}-1]} .$$

The solution of this equation is obtained by taking its inverse z transformed, as developed in Appendix A, yielding

$$R_s(k) = (1-\alpha)^k e^{-bkT} R_o - \frac{\alpha}{(1-\alpha)e^{-bT}-1}\,[1-(1-\alpha)^k e^{-bkT}] \qquad (11)$$

Using the same method for Equation (9), we take the z transform to get

$$zR(z) - zR(1) = (1-\alpha)e^{-bT}R(z) + \alpha e^{-bT}\,\frac{z}{z-1}, \qquad (12)$$

or

$$[z-(1-\alpha)e^{-bT}]R(z) = zR(1) + \frac{\alpha e^{-bT}z}{z-1} .$$

Solving for R(z), we get

$$R(z) = \frac{z}{z-e^{-bT}(1-\alpha)} R(1) + \frac{\alpha e^{-bT}z}{(z-1)[z-(1-\alpha)e^{-bT}]}$$

which expanded in partial fractions yields

$$R(z) = \frac{z}{z-e^{-bT}(1-\alpha)} R(1) + \frac{z}{(z-1)} \frac{\alpha e^{-bT}}{[1-(1-\alpha)e^{-bT}]}$$

$$+ \frac{z}{[z-(1-\alpha)e^{-bT}]} \frac{\alpha e^{-bT}[(1-\alpha)e^{-bT}]}{[(1-\alpha)e^{-bT}-1]} .$$

Taking the inverse z transform, we find the solution in the time domain to be

$$R(k) = (1-\alpha)^k e^{-bkT} R(1) - \frac{\alpha e^{-bT}}{(1-\alpha)e^{-bT}-1}[1-(1-\alpha)^k e^{-bkT}] \qquad (13)$$

Equation (13) gives the reliability of an item immediately before the rework at time t = kT for a given initial condition R(1), the reliability of the item just before the first rework. Equation (11) gives the reliability of the item after the rework at time t = kT for a given initial condition R_o, the reliability of the item at the time of

acquisition. Comparing these two equations we notice that they are different only by a factor of e^{-bT}, as might have been expected from the Expression (6) that relates these two reliabilities.

As an example, values of $R_s(k)$ and $R(k)$ are shown in Table (I) for various values of R_o, α, b and T. As can be seen in Table I, for T = 400 and $R_o = 0.8$ we have, for $\alpha = 0.8$, the case "not as good as new"; for $\alpha = 0.569$ the case "as good as new"; and for $\alpha = 0.7$, an improvement of item reliability constituting the "better than new" case. The value of α that leads to the case "as good as new" can be found by setting the reliability following the first rework $R_s(1)$ equal to the initial reliability R_o. Thus from equation (11) we have

$$R_s(1) = (1-\alpha)e^{-bT}R(0) - \frac{\alpha}{(1-\alpha)e^{-bT}-1}[1-(1-\alpha)e^{-bT}] = R(0),$$

which, rearranged, gives

$$\alpha = [1 - (1-\alpha)e^{-bT}]\ R(0)$$

or

$$\alpha - \alpha R(0)e^{-bT} = R(0)\ (1-e^{-bT}).$$

Solving for α yields

TABLE I

Examples of reliability immediately before rework $R(k)$ and reliability following rework $R_s(k)$.
Initial reliability = R_o
Rework Effectiveness = α
Deterioration parameter = b
Turnover period = T

$R_o = 0.8$ T = 400

	$\alpha = 0.2$		$\alpha = 0.569$		$\alpha = 0.7$	
k	R(k)	$R_s(k)$	R(k)	$R_s(k)$	R(k)	$R_s(k)$
1	0.536	0.629	0.536	0.800	0.536	0.861
2	0.422	0.537	0.536	0.800	0.577	0.873
3	0.360	0.488	0.536	0.800	0.585	0.876
4	0.327	0.462	0.536	0.800	0.587	0.876

$R_o = 0.8$ $\alpha = 0.7$

	T = 50		T = 100		T = 250	
k	R(k)	$R_s(k)$	R(k)	$R_s(k)$	R(k)	$R_s(k)$
1	0.761	0.928	0.724	0.917	0.623	0.887
2	0.883	0.965	0.830	0.949	0.691	0.907
3	0.918	0.975	0.859	0.958	0.701	0.912
4	0.928	0.978	0.866	0.960	0.710	0.913

$R_o = 0.95$ T = 100

	$\alpha = 0.07$		$\alpha = 0.8$		$\alpha = 0.9$	
k	R(k)	$R_s(k)$	R(k)	$R_s(k)$	R(k)	$R_s(k)$
1	0.860	0.958	0.860	0.972	0.860	0.986
2	0.867	0.960	0.979	0.976	0.892	0.989
3	0.869	0.961	0.883	0.977	0.895	0.990
4	0.869	0.961	0.884	0.977	0.895	0.990

b = 0.001 for all cases

$$\alpha = \frac{R(0)\ (1-e^{-bT})}{1 - R(0)e^{-bT}} \tag{14}$$

D. THE CONTROL VARIABLE

As pointed out by Bohannan [2], with these values a steady state in reliability is reached by the fourth rework and the influence of the initial reliability is lost. Thus in order to rework an item to achieve a desired reliability, we have to choose values for the rework effectiveness α, or the turnover period T or both, so that the steady state is not below the desired value. The turnover period can be shortened by increasing the rework rate ρ or by reducing the inventory size N. Another way to have a higher reliability would be by improving the storage environment, which would reduce the reliability deterioration rate b, if this is possible. However, the inventory size N and the deterioration rate b are not generally considered part of a rework policy. In our formulation only the rework rate ρ and the rework effectiveness α could be changed within a rework policy.

In this thesis we choose to work with a fixed rework rate ρ and to use the rework effectiveness α to control the item reliability. This appears to be the usual practice in preventive maintenance involving a fixed schedule and several levels of maintenance or rework (corresponding to several values of α). The use of the rework rate ρ to

control the system makes the system non linear, involving equations that are time variant. This can be solved by another area of the control theory which while not covered in this thesis, can be done in a further work.

In this chapter we derived a rework model for the case when we have a constant rework rate. We solved our discrete recursive equations for the reliability before the rework, R(k), and after the rework, $R_s(k)$, by the z transform technique. The result is in a closed form rather than in the form of a summation, making this solution more suitable for further analysis as will be done in Chapter IV. Before this analysis, we will first derive in the next chapter this same closed form solution by applying the control theory approach.

III. DEVELOPMENT OF THE CONTROL MODEL

In Chapter II we developed the rework model and found the system of first order difference equations relating the reliability immediately before the rework R(k) to the reliability after the rework $R_s(k)$:

$$R(k+1) = R_s(k)\, e^{-bT}, \tag{15}$$

and

$$R_s(k) = \alpha(1-R(k)) + R(k). \tag{16}$$

We then solved this system using the z transform technique, obtaining an expression for the reliability after any rework as a function of the initial reliability, R_o,

$$R_s(k) = (1-\alpha)^k e^{-bkT} R_o - \frac{\alpha}{(1-\alpha)e^{-bT}-1}[1-(1-\alpha)^k e^{-bkT}], \tag{17}$$

and an expression for the reliability immediately before any rework as a function of this reliability before the first rework:

$$R(k) = (1-\alpha)^k e^{-bkT} R(1) - \frac{\alpha e^{-bT}}{(1-\alpha)e^{-bT}-1}[1-(1-\alpha)^k e^{-bkT}] \tag{18}$$

In this chapter we will introduce a control theory approach to the rework problem, and then derive these same expressions in terms of control theory. This will provide a basis for later solving the problem of obtaining an optimal sequence of rework levels.

A. THE CONTROL THEORY APPROACH

In this section we will review, in a simplified form, the main aspects of control theory that are of interest to our rework model.

The first fundamental concept underlying a control system is that of a dynamic model, that is a model describing the relationships among the relevant variables and parameters of our system. In the model, the variables are allowed to change with time in a deterministic way, and the system behavior depends not only on their values at any instant, but also on their past values and the rate of change of these variables with time. The equations derived in Chapter II constitutes such a dynamic model of our system.

The second fundamental concept we will need is that of a closed-loop control system, i.e., one in which the output has a direct effect upon the control action, as shown schematically in Figure 2. The output and the reference input are compared in the controller and we say that the output is fed back to the controller. The difference between the reference signal and the feedback signal, called the error signal or input signal to the plant, is then used to

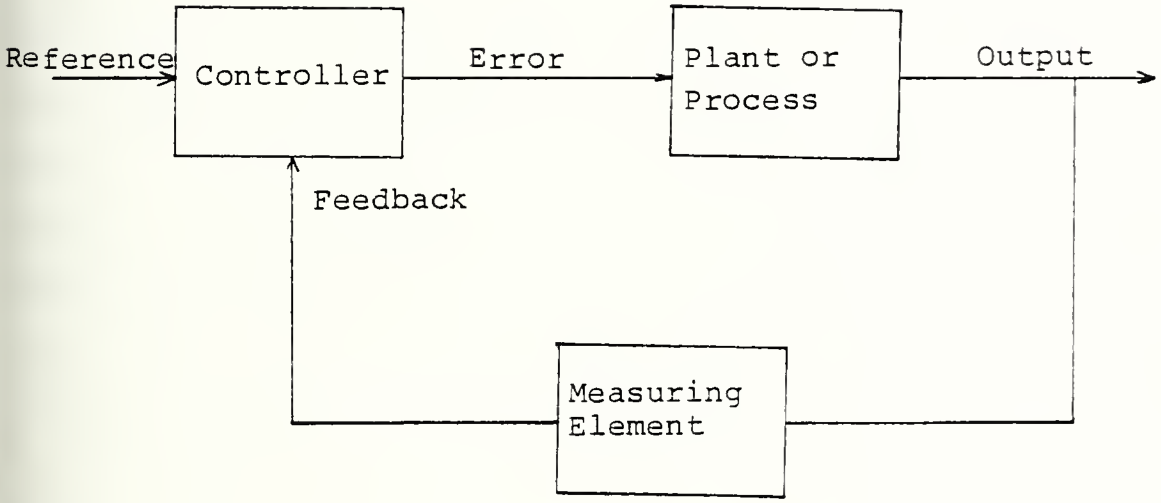

FIGURE 2. Block Diagram of a Control System

change the behavior of our plant or process with the goal of bringing the output of the system to a value that gives a zero error.

The "plant" is any physical or abstract object to be controlled. Sometimes we feedback a measure of our output: for this case, a box representing the measurement device would be drawn in the feedback branch of the block diagram of Figure 2. In summary, the term "closed loop" implies the use of feedback action in order to reduce system error. An open loop control system is a control system in which the output has no effect upon the control action, that is, the output is neither measured nor fed back for comparison with the reference. An advantage of the closed-loop control system is that the use of feedback makes the system output relatively insensitive to external disturbances and internal variations in system parameters, since a correction action is taken anytime the error is not zero.

An useful tool in control theory is a signal flow graph. This is a pictorial representation of a set of simultaneous algebraic equations in which each variable is represented by a graphical symbol called a node, and the dependencies between pairs of variables are represented by directed branches drawn between pairs of nodes. These dependencies between two variables are called transfer functions, or gains and are defined as the ratio of the incoming variable to the variable at the end of the branch. Transfer functions

are used to label the branches, and indicate that a multiplication operation is done upon the value entering the branch in the arrow direction, delivering a new branch value to the node where the branches terminates. The nodes are also summing devices which sum all values arriving by the way of incoming branches. As an example of this, Figure 3 shows the signal flow graph for a hypothetical system defined by the set of equations

$$x_1 = bx_2 + dx_4 + fu$$

$$x_2 = ax_1$$

$$x_3 = ix_2 + ex_4 + gu$$

$$x_4 = cx_3$$

$$Y = hx_4$$

Given the signal flow graph of a system we often wish to write the overall transfer function from an input or reference value to the output. This can be obtained directly from the signal flow graph by inspection or by use of Mason's Theorem. This theorem, which we will use later, expresses the transfer function in terms of various loop gains and the parallel gains from input to output, and

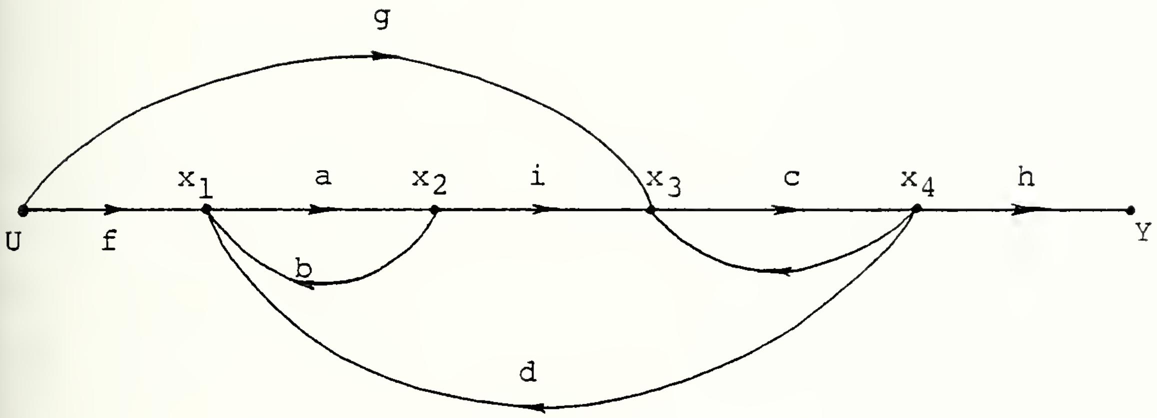

FIGURE 3. Example of a signal flow graph diagram

states that the transfer function from input u to a response Y is

$$T = \frac{Y}{U} = \frac{\sum_i P_i \Delta_i}{\Delta}$$

where the terms are defined as:

(1) Δ is the determinant of the feedback configuration and is calculated from the equation

$$\Delta = 1 - \sum L_j + \sum' L_k L_\ell - \sum' L_m L_n L_o + \dots ,$$

where the L_j (or L_k, etc.) are loop gains all the way around a feedback loop in the system. Thus $\sum L_j$ means the sum of all loop gains. The next term $\sum' L_k L_\ell$ is the sum of all products of pairs of different loop gains - e.g., $L_1 L_3$ and so forth. The prime on the summation means we use only the products for pairs of gains of non-touching loops. In other words, $L_1 L_3$ is included only if loop 1 does not touch loop 3 (two loops touch if they have at least one node in common). Likewise, $\sum' L_m L_n L_o$ is the sum of all products three at a time, where again each of the three loops does not touch the other two.

(2) P_i is a path gain of the ith forward path from input U to output Y (a path which contains no loops).

(3) Δ_i is the system determinant Δ after we have excluded all loops which touch the P_i path.

The Mason's Theorem is better understood by an example. Suppose our system is described by the signal flow graph of Figure 3 and we want to determine the overall transfer function Y/U. The loop gains are

$$L_1 = ab , \quad L_2 = ce , \quad L_3 = aicd .$$

Hence

$$\sum L_j = ab + ce + aicd .$$

The loop L_3 touches both L_1 and L_2 but L_1 and L_2 are non-touching. Hence

$$\sum{}' L_k L_\ell = aicd ,$$

and there are no three non-touching loops. We can then write the determinant Δ

$$\Delta = 1 - \sum L_j + \sum{}' L_k L_\ell ,$$

or

$$\Delta = 1 - (ab + ce + aicd) + aicd .$$

The direct paths from U to Y are

$$P_1 = f\,a\,i\,c\,h$$

and

$$P_2 = g\,c\,h\ .$$

Since P_1 touches all loops,

$$\Delta_1 = 1.$$

Path P_2 does not touch L_1, hence

$$\Delta_2 = 1 - L_1 = 1 - ab\ .$$

Substituting into Mason's formula, we have

$$T = \frac{Y}{U} = \frac{P_1\Delta_1 + P_2\Delta_2}{\Delta},$$

or

$$T = \frac{f\,a\,i\,c\,h + g\,c\,h(1 - ab)}{1 - (ab + ce + aicd) + aicd}\ .$$

Summarizing, we can say that Mason's theorem gives a simple, fast procedure for writing a desired transfer function directly from the signal flow diagram. The response of our system can then be immediately derived from this transfer function as proportional to the input,

$$Y = T\,U\,,$$

and this is called the solution to our control model, i.e., an expression for the response of our system as a function of any input to the system and a known (derived by Mason's Rule) overall transfer function.

B. THE RELIABILITY CONTROL MODEL

Our concept of a system is not limited to physical systems. The concept can be applied to abstract, dynamic phenomena such as those encountered in economics and operations research. Also, feedback control systems are not limited to the field of engineering but can be of particular interest to the manager, public official, operations researcher, biologist, and design engineer. In this section we will structure the reliability model in control theory terms. The _plant_ of the control theory approach will be the inventory, where the (decreasing) reliability of items is our variable of interest, or _state variable_. The controller will be substituted by a decision-making function that will decide upon a level of maintenance which is optimal in the sense of minimizing a certain performance measure. We will defer optimization until Chapter VI, and for the present will assume a single level of rework, as defined in Section D of Chapter II, corresponding to a certain effectiveness of the rework α.

Our dynamic model is described by expressions (15) and (16),

$$R(k+1) = R_s(k)\, e^{-bT}, \quad k = 0,1,2, \ldots,$$

and

$$R_s(k) = \alpha(1 - R(k)) + R(k), \quad k = 1,2,3, \ldots,$$

and the signal flow graph can be drawn in several ways, one of which is shown in Figure 4. At the second node from the left the present item reliability, whose measured value is fed back to this point, is compared with the reference value, R_D, that is made here equal to 1.0 due to our definition of rework effectiveness (this will be better clarified later). The difference, when not zero, produces an automatic decision to rework the item, and after the rework, this difference becomes multiplied by α since from expression (16) we have

$$R_s(k) - R(k) = \alpha(1 - R(k)) = U(k) \qquad (19)$$

This difference will be denoted by U(k), and called the input to our plant. Physically the item is returned to the inventory at this point and the reliability of the item is given by the sum

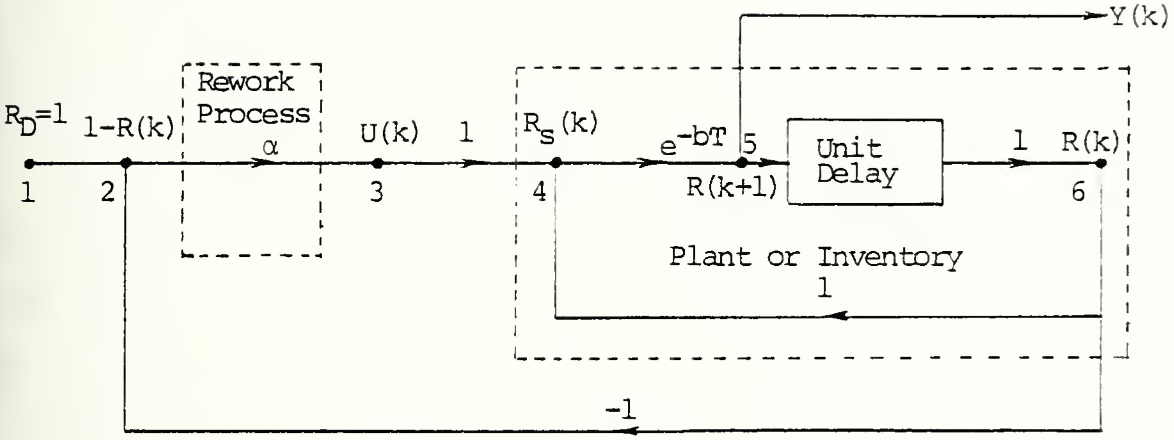

FIGURE 4. Reliability Flow Graph for the Rework Model

$$R_s(k) = U(k) + R(k) .$$

This sum is performed at the fourth node. While in the inventory the item reliability will deteriorate by a factor e^{-bT} given by expression (15) and at the end of a period of length T we have the item reliability R(k+1) immediately after the (k+1)th rework represented at the fifth node by R(k+1). The relation between R(k) and R(k+1) is a function called a unit delay of time, represented by a box in the flow graph, with the meaning that the output of our system R(k+1) is one period T ahead of the value fed back, R(k), used to compute it.

The variable Y(k) is used to emphasize the fact that R(k+1) is the output of our control system. Branches with a gain equal to 1.0 are used when there are variables that are just renamed or going to be operated on in another mode.

Taking the z transform on both sides of Equations (15) and (16) by applying the properties given in Appendix A, we have

$$R(z) = z^{-1} R_s(z) e^{-bT} + R(1) , \tag{20}$$

and

$$R_s(z) = \alpha(\frac{z}{z-1} - R(z)) + R(z) , \tag{21}$$

and the signal flow graph can be drawn as in Figure 5.

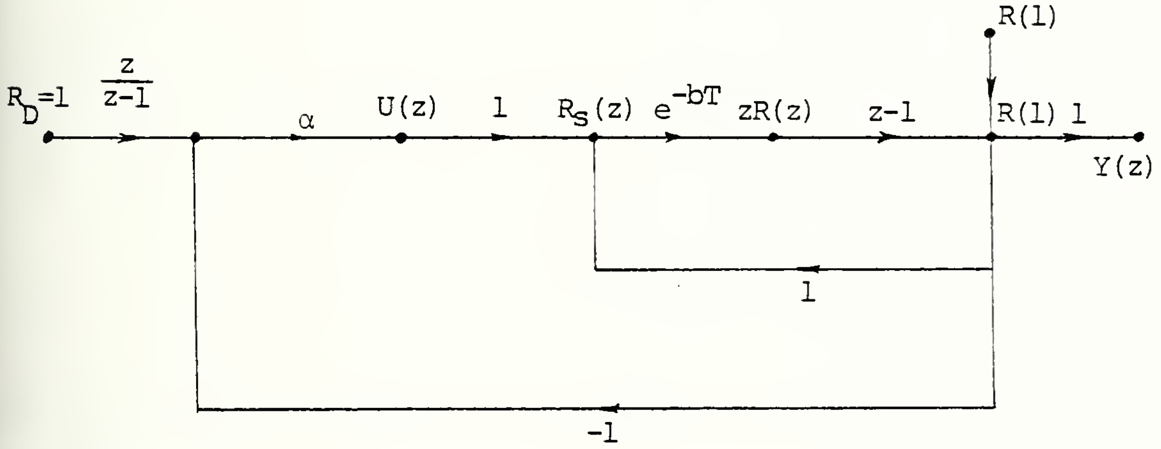

FIGURE 5. Reliability Signal Flow Graph to the Control Model in the z Domain

The response of the system, Y(z), can be now obtained by applying Mason's theorem as shown earlier. Since we can consider that we have here two input values, namely R(1) and R_D, this yields the overall transfer function with respect to the initial condition R(1) as

$$\frac{Y(z)}{R(1)} = \frac{1}{(1 - \alpha e^{-bT} z^{-1} + e^{-bT}} = \frac{z}{z-(1-\alpha)e^{-bT}} ,$$

and with respect to the reference value R_D,

$$\frac{Y(z)}{R_D} = \frac{\alpha e^{-bT}(\frac{z}{z-1})z^{-1}}{1 - \alpha e^{-bT} z^{-1} + e^{-bT}} = \frac{\alpha e^{-bT} z}{(z-1)(z-(1-\alpha)e^{-bT})} .$$

The response of the system due to the reference value that here is made equal to 1.0 is then

$$Y(z) = \frac{R_D \alpha e^{-bT} z}{(z-1)(z-(1-\alpha)e^{-bT})} ,$$

and the response of the system due to the initial condition is given by

$$Y(z) = \frac{R(1)z}{(z-(1-\alpha)e^{-bT})} .$$

Using the superposition principle for differential equations, the total response is

$$Y(z) = \frac{\alpha e^{-bT} z}{(z-1)(z-(1-\alpha)e^{-bT})} + \frac{R(1)z}{(z-(1-\alpha)e^{-bT})} \quad (22)$$

This is the same as the z transform of Equation (13) in Chapter II, and the inverse z transform of the total response then gives the same Equation (13) for the reliability of an item immediately before a rework. This is

$$Y(k) = R(k) = (1-\alpha)^k e^{-bkT} R(1) - \frac{\alpha e^{-bT}}{(1-\alpha)e^{-bT}-1}[1-(1-\alpha)^k e^{-bkT}],$$

$$k = 1,2,3, \ldots ,$$

where the argument (k+1) was changed to k. In this case the z transform had to be applied because the unit delay present in the system, does not provide a linear function necessary for the application of Mason's theorem. In the z domain, however, the unit delay becomes a very simple function and the solution may be found easily.

If the reference input R_D is not equal to 1.0 but is the desired value of the item reliability that we want $R_s(k)$ to match after some reworks then the definition of rework effectiveness would become

$$\alpha = \frac{R_s(k) - R(k)}{R_D - R(k)} .$$

Here, if the reliability immediately before a rework was greater than the value of the desired reliability, the value of α would be negative. To avoid this problem we might use a decision rule that could be expressed as a function $M(R,R_D)$ defined as

$$M = \begin{cases} 0 & \text{if} \quad R_D - R(k) < 0 \\ \alpha & \text{if} \quad R_D - R(k) > 0 \end{cases} \tag{23}$$

with the meaning that no rework has to be done when the item reliability R(k) is greater than the desired value R_D. Our model would then become

$$R_s(k) = R(k) + M(R_D - R(k)) \tag{24}$$

This approach, however, would introduce a non-linearity into our system that could not be eliminated by the z transform procedure, and the solution of our problem would probably have to be obtained by computer. Hopefully we can handle this problem and find an optimal sequence of reworks that makes the value of $R_s(k)$ to match the desired value R_D by using the optimization process that will be developed in Chapter V.

Summarizing, we have structured the item reliability model in terms of control theory for periodic reworks with effectiveness of rework α as defined by equation (1). We are now able to apply the tools of control theory to find: (i) the limits of stability of our system as a function of the rework period T, (ii) the stable final value of the reliability as the number of reworks increase, and the use of a Kalman filter. This will be shown in the next chapter.

IV. APPLICATIONS OF THE CONTROL THEORY MODEL

In this chapter we will show how the results of our control model can be applied to study the behavior of the item reliability. First, we will study the stability of our system, and then the final value reliability when t increases. After introducing randomness to the system, we will show the use of a Kalman Filter to estimate the value of the reliability that is fed back.

A. SYSTEM STABILITY

A linear control system is stable if the output eventually comes back to an equilibrium state when the system is subjected to a disturbance. In our reliability system we are interested in knowing if there is any range of values of the rework effectiveness factor α for which the system becomes unstable, i.e., the reliability might continue to decrease until the items need to be replaced rather than reworked.

From control theory we know that the characteristic equation of a control system is equal to the denominator of the overall transfer function plus 1.0 and that for the system to be stable the roots of the characteristic equation, in the z domain, have to be inside the unity circle [10]. For our reliability system we have from Equation (21) the characteristic equation in the z domain as

$$(z-1)[z-(1-\alpha)e^{-bT}] + 1 = 0 .$$

Rearranging our characteristic equation, we have

$$z^2 - [1+(1-\alpha)e^{-bT}]z + [1+(1-\alpha)e^{-bT}] = 0 ,$$

with complex roots at

$$z = \frac{1+(1-\alpha)e^{-bT}}{2} \pm j\,\frac{1}{2}\sqrt{[1+(1-\alpha)e^{-bT}]^2-4[1+(1-\alpha)e^{-bT}]} .$$

The absolute value of these complex roots is given by

$$|z| = \sqrt{\tfrac{1}{2}[(1-\alpha)^2e^{-2bT}-1]} .$$

Here,

$$|z| < 1$$

or

$$|z|^2 < 1$$

when

$$\tfrac{1}{2}[(1-\alpha)^2e^{-2bT}-1] < 1$$

or

$$(1-\alpha)^2 < 3e^{2bT}$$

This gives a second order inequality in α

$$\alpha^2 - 2\alpha + 1 - 3e^{2bT} < 0$$

that when solved shows us that for a stable system, the value of α must be within the range

$$1-e^{bT}\sqrt{3} < \alpha < 1+e^{bT}\sqrt{3} \qquad (25)$$

For the example of Chapter II with $b = 0.001$ and $T = 100$, we get

$$0.56 < \alpha < 2.56$$

that agrees well with the value of α in Table I for which the item is "as good as new" after each rework.

Outside these limits or when α is less than 0.56, (since α by definition is less than unity), the system is unstable. Here, the item will have its reliability deteriorating with time until the item needs to be replaced rather than reworked, i.e., for $\alpha < 0.56$ eventually $R \to 0$.

B. THE FINAL VALUE IN STABLE SYSTEMS

When the system is stable its response tends to a constant value as the time increase whenever there is no

disturbance input. To find this steady state value we will apply the final value theorem, derived in Appendix A, that gives us the limit of the system response when time becomes very large. The theorem is stated as

$$\lim_{k \to \infty} Y(k) = \lim_{z \to 1} [(z-1)Y(z)] .$$

For our overall transfer function, this results in

$$\lim_{k \to \infty} R_s(k) = \lim_{z \to 1} \left[\frac{\alpha z}{z-(1-\alpha)e^{-bT}} + \frac{zR(0)(z-1)}{z-(1-\alpha)e^{-bT}}\right]$$

$$= \frac{\alpha}{1-(1-\alpha)e^{-bT}} \tag{26}$$

Substituting selected values from Table 1, we get

$\alpha = 0.7$ $\quad T = 100$ $\quad \lim_{t \to \infty} R_s(k) = 0.961$

$\alpha = 0.8$ $\quad T = 100$ $\quad \lim_{t \to \infty} R_s(k) = 0.977$

$\alpha = 0.7$ $\quad T = 400$ $\quad \lim_{t \to \infty} R_s(k) = 0.876$

and these limits agree well with the values taken from the table for the fourth rework.

As we can clearly conclude from expression (26) the steady state value of the reliability after the rework $R_s(\infty)$

depends only on the rework effectiveness α and on the turnover period T for a given degradation parameter b, and is thus independent of the initial value of the reliability as was intuitively observed in Chapter II.

C. ERROR COMPENSATION USING A KALMAN FILTER

In this section we will show how a Kalman Filter can be applied to our model in order to compensate for errors in the measurement of the item reliability. Again, measurement here does not imply a physical action. We suppose the nature of the stored item is such that it receives frequent diagnostic checks which yield an estimate of its reliability, and such "measurements" are subject to random error.

In the model developed in Chapter III we have shown how the item reliability is fed back so that a decision can be made about doing a rework so that the item's reliability can approach a desired value. In practice we have to measure the item reliability or estimate it by some way in order to feed back this observed value. This measurement process, however, contributes to the variance within the system. Other sources of variance in the process include variability in the item environment and in subsystems reliabilities due to repairing or substitution of parts. We call these random errors as noisy or random input to our plant, since we can model these errors as random variables that are input somewhere in the plant as will see later.

If we do not want our control system following these random inputs excessively it would be a good policy to smooth the reliability values over time and then feed back a predicted value upon which we made our decisions.

A useful, smoothing device is a Kalman Filter, and a block diagram of a general system with a Kalman filter is shown in Figure 6. Other filters could be used here but with a discrete system we chose the Kalman filter: it is the optimum recursive filter in the sense that it minimizes the variance of the estimator error. The random input at the plant in Figure 6 is designated by the random variable $\omega(k)$ and represents the error introduced by the environment and rework process. The random error in the measurement is designated by the random variable $v(k)$.

Making the assumption that these noises are additive expected values, we can restate our system equations (15) and (16) as

$$R(k+1) = e^{-bT} R_s(k) + w(k)$$

and

$$R_s(k) = (1-\alpha) R(k) + \alpha R_D,$$

and substituting the second equation into the first, we have

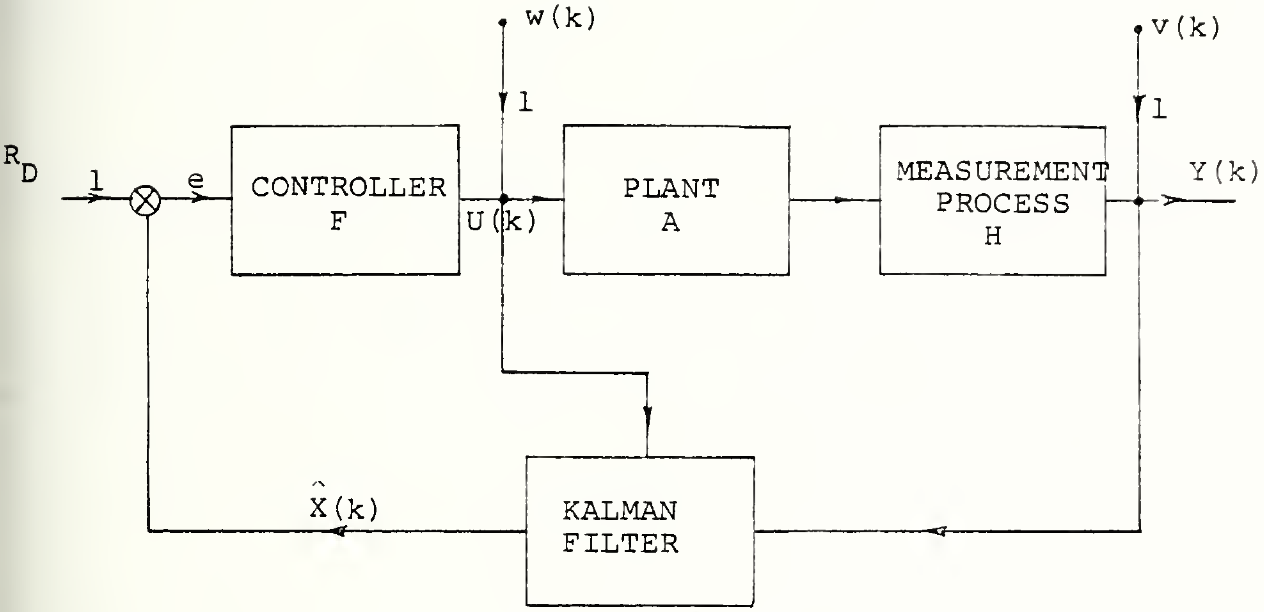

FIGURE 6. Block diagram of a general control system with Kalman Filter

$$R(k+1) = e^{-bT}[(1-\alpha)R(k)+\alpha R_D] + w(k) .$$

Rearranging and grouping terms we have

$$R(k+1) = e^{-bT}R(k) + e^{-bT}U(k) + w(k) \qquad (27)$$

where

$$U(k) = \alpha[R_D - R(k)]$$

as before and $R_D = 1$.

If we introduce the Kalman Filter in our system we are feeding back an estiamted value $\hat{R}(k)$ instead of the actual value R(k) and the input to the plant becomes

$$U(k) = [R_D - \hat{R}(k)] \qquad (28)$$

The measurement process can be modelled by the equation

$$Y(k) = H\ R(k) + v(k) \qquad (29)$$

where H can be any function describing the measurement, and v(k) is the random variable in the measurement process.

We need the following additional assumptions to make the application of a Kalman Filter valid:

(a) The measurement error v(k) and the random process input w(k) has zero mean and are uncorrelated between states, with variances S and Q respectively, or

$$E[v(k)v(j)] = S\,\delta_{kj}\ , \quad k,j = 0,1,2,\ \ldots\ ,$$

$$E[w(k)w(j)] = Q\,\delta_{kj}\ , \quad k,j = 0,1,2,\ \ldots\ ,$$

where δ_{kj} is the Kronecker delta function.

(b) The initial state is a random variable which has known mean $E[R(0)] = R_o$ and variance M.

(c) The estimator is characterized by the linear relationship

$$\hat{R}(k|k) = \hat{R}(k|k-1) + G(k)[Y(k) - H\hat{R}(k|k-1)]\ , \quad k = 0,1,2,\ldots,$$

where:

$\hat{R}(k|k)$ is the optimal estimate of R(k) given observations at times up to and including k,

$\hat{R}(k|k-1)$ is the optimal prediction of R(k) given observations at times up to and including k-1, and

G(k) is the gain of the Kalman Filter at each step.

(d) The random input and initial state are uncorrelated.

(e) The random errors in the plant process or random input w(k) and the random error in the measurement process v(k) are uncorrelated or

$$E[w(k)v(j)] = 0 \qquad j,k = 0,1,2,\ \ldots\ .$$

The functioning of the Kalman filter is by predicting the value of the item reliability $\hat{R}(k|k-1)$ using the estimated value at the end of the (k-1)th period, multiplying this predicted value by the function H, so that this new value can be directly compared with the measured value. The filter gets the correction term $[Y(k)-H\hat{R}(k|k-1)]$ that can be conveniently weighted by the gain G(k) to correct the predicted value, and obtains a new estimated value for the kth period.

Notice that if we did not use a predicted value, what is equivalent to say that $\hat{R}(k|k-1) = \hat{R}(k-1|k-1) = \hat{R}(k-1)$, we would have exponential smoothing and the gain G(k) would then be the smoothing constant. With the predicted value any trend is recursively incorporated in the estimated value.

The best prediction $\hat{R}(k|k-1)$ that can be made is clearly by using our model (Equation (27)),

$$\hat{R}(k|k-1) = e^{-bT}\hat{R}(k-1|k-1) + e^{-bT}U(k-1), \qquad (30)$$

and from assumption (c) above, the equation for the estimated value $\hat{R}(k|k)$ is

$$\hat{R}(k) = \hat{R}(k|k) = \hat{R}(k|k-1) + G(k)[Y(k)-H\hat{R}(k|k-1)]. \qquad (31)$$

With these two equations and the assumptions above it is shown [9] that the sequence of gains that minimizes the

variance of the estimator error $P(k|k)$ is given by the recursive equations

$$G(k) = P(k|k-1)H[H^2P(k|k-1)+S]^{-1} , \tag{32}$$

$$P(k|k) = [1 - HG(k)]\ P(k|k-1) , \tag{33}$$

and

$$P(k|k-1) = (e^{-bT})^2 P(k-1|k-1) + Q. \tag{34}$$

These are initialized by the value

$$P(0|-1) = M . \tag{35}$$

From these equations we can see that the Kalman Filter gain $G(k)$ does not depend on the measurement values $Y(k)$, and can thus be computed in advance and stored for later use when processing the measurements as they become available. We can see also that the gain at time k, $G(k)$, is inversely realted to the variance of measurement error S — the more uncertainty in the measurements, as reflected by a larger S, the smaller $G(k)$ will be, and the less the correction term $[Y(k)-H\hat{R}(k|k-1)]$ in Equation (31) will be weighted in determining the next estimate. The random input error Q also affects the gain $G(k)$, and as the uncertainty in the

prediction of $P(k|k-1)$ increases and the more uncertainty we have in our model, the larger will be $G(k)$. For large values of M or no confidence in the initial guess $\hat{R}(0|-1)$ the filter makes $\hat{R}(0|0)$ equals to the first measurement $Y(0)$ since $G(0)$ is equal to 1.0. When the initial guess is better the filter makes a weighted combination of the guess $\hat{R}(0|-1)$ and the first measurement $Y(0)$, as one can conclude from Equation (31).

A flow graph in the time domain of our system is shown in Figure 7. The lower part of the flow graph is the Kalman Filter, which receives as input the measured value $Y(k)$ and the input to the plant $U(k)$, and has as output the estimated value $\hat{R}(k)$. This is multiplied by the measurement function H and the result is compared with the observed value $Y(k)$. The difference is then multiplied by the gain $G(k)$ and summed to the predicted value to produce the estimated value, as given by Equation (31).

When we have more than one state variable, an advantage of the Kalman filter not shown here is that we do not need to measure all state variables but only a smaller number to get all the estimated values we need to feed back to the plant.

As an example, suppose we have values from Table I in that $\alpha = 0.7$, $b = 0.001$, $T = 100$ and $\hat{R}_o = 0.8$; suppose also that the variance in the estimated value of R_o is $M = 0.1$, and that the variances in the random input and in the measurement are known to be $Q = 0.1$ and $S = 0.1$. Let our measurement equation be given by

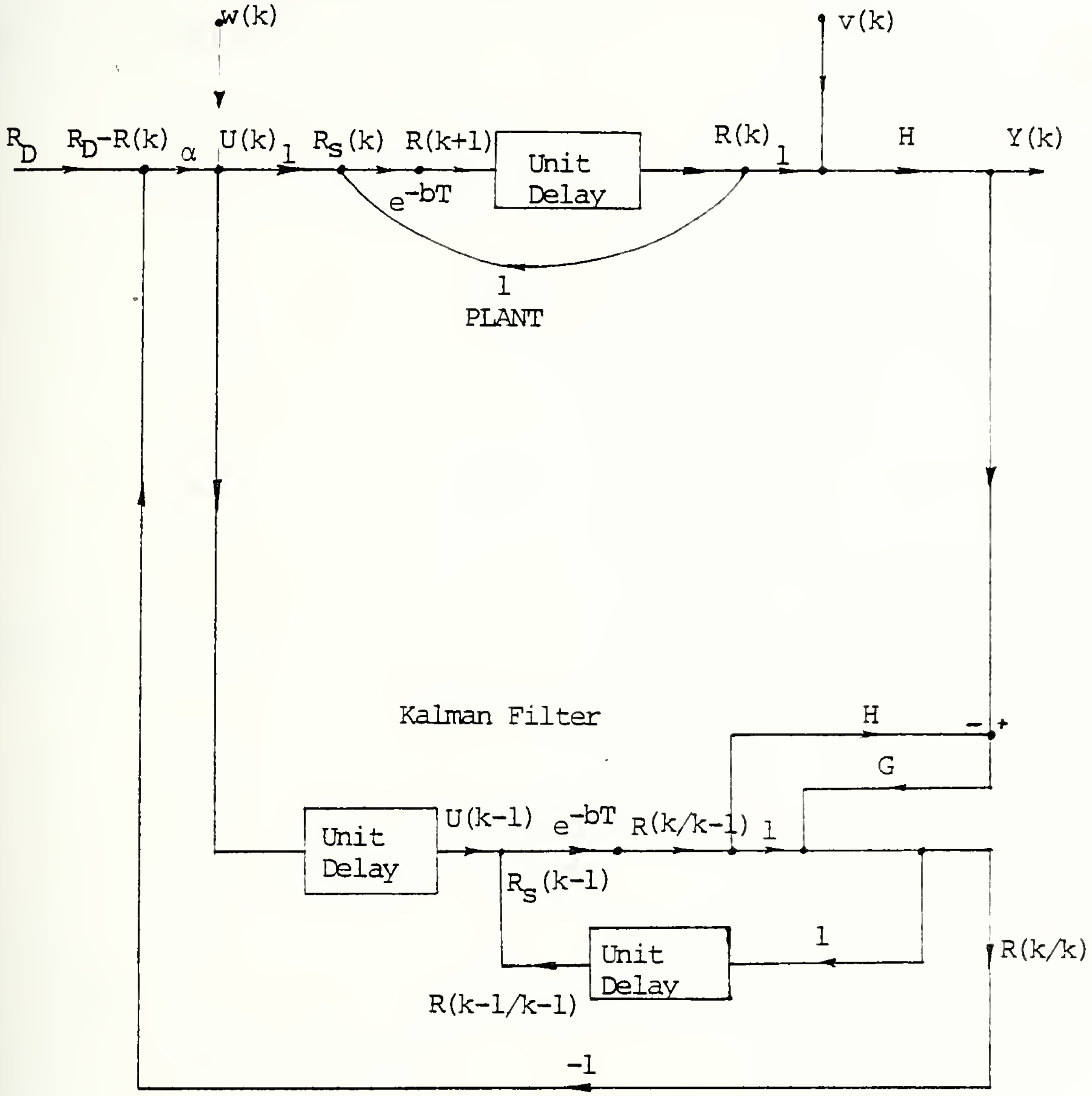

FIGURE 7. Flow graph in the time domain for the rework system with a Kalman Filter

$$Y(k) = R(k) + v(k)$$

which in terms of Equation (29), implies $H = 1$. With these values we can compute the Kalman filter gain for four periods using Equations (32) to (35):

$$G(0) = 0.5,$$

$$G(1) = 0.58,$$

$$G(2) = 0.60,$$

$$G(3) = 0.60, \text{ and}$$

$$G(4) = 0.60.$$

Suppose the measured values are

$$Y(1) = 0.75,$$

$$Y(2) = 0.82,$$

$$Y(3) = 0.86, \text{ and}$$

$$Y(4) = 0.85.$$

and we guess a value for the initial reliability $\hat{R}(1|0) = 0.724$. We obtain, by using Equations (30) and (31), the values

$$\hat{R}(1) = 0.737 ,$$

$$\hat{R}(2) = 0.825 ,$$

$$\hat{R}(3) = 0.859 , \text{ and}$$

$$\hat{R}(4) = 0.856 .$$

Notice that the reliability before the rework, $R(k)$, is not defined for $k = 0$. For this reason we use as initial guess $\hat{R}(1|0)$ the value of the reliability before the first rework computed from the initial value $R_o = 0.8$ and use the gain $G(0)$ to estimate $\hat{R}(1)$, the gain $G(1)$ to estimate $\hat{R}(2)$ and so on. The gain would not be delayed by one period if we had decided to call the reliability before the first rework by $R(0)$.

D. EXTENSIONS TO MORE THAN ONE DIMENSION

In a missile we can consider for example its booster, cruise motor, guidance, and warhead as four subsystems that are independently influenced by the environment and have different reliability deterioration factor b. To extend our model for this case we need to decompose our system in several distinct subsystems. Here we may put the equations for the subsystems in matrix form.

We can define the vector of reliabilities of the subsystems that will be called our state variables, as

$$\underset{\sim}{R}(k) = \begin{bmatrix} R_i(k) \\ R_2(k) \\ R_3(k) \\ R_4(k) \end{bmatrix} ,$$

and a vector of reference reliabilities as

$$\underset{\sim}{R}_D = \begin{bmatrix} R_{D1} \\ R_{D2} \\ R_{D3} \\ R_{D4} \end{bmatrix} .$$

The input vedtor can now be defined as before in terms of the effectiveness of the rework α_i for each subsystem:

$$\underset{\sim}{U}(k) = \begin{bmatrix} U_1(k) \\ U_2(k) \\ U_3(k) \\ U_4(k) \end{bmatrix} = [\alpha_1 \ \alpha_2 \ \alpha_3 \ \alpha_4] \begin{bmatrix} R_{D1} - R_1 \\ R_{D2} - R_2 \\ R_{D3} - R_3 \\ R_{D4} - R_4 \end{bmatrix}$$

or

$$\underset{\sim}{U}(k) = \underset{\sim}{\alpha}^T [\underset{\sim}{R}_D - \underset{\sim}{R}(k)] .$$

We can then write our state equation for the system with random input as

$$\underset{\sim}{R}(k+1) = \underset{\sim}{\Phi}\ \hat{\underset{\sim}{R}}(k) + \underset{\sim}{\phi}\ U(k) + \underset{\sim}{w}(k)\ , \tag{36}$$

where the matrix $\underset{\sim}{\phi}$ is

$$\underset{\sim}{\phi} = \begin{bmatrix} e^{-b_1T} & 0 & 0 & 0 \\ 0 & e^{-b_2T} & 0 & 0 \\ 0 & 0 & e^{-b_3T} & 0 \\ 0 & 0 & 0 & e^{-b_4T} \end{bmatrix}$$

The measurement equation is

$$\underset{\sim}{Y}(k) = \underset{\sim}{H}\ \underset{\sim}{R}(k) + \underset{\sim}{v}(k)\ , \tag{37}$$

and the vector of estimated values is given by

$$\hat{\underset{\sim}{R}}(k) = \hat{\underset{\sim}{R}}(k|k) = \hat{\underset{\sim}{R}}(k|k-1) + \underset{\sim}{G}(k)\,[\underset{\sim}{Y}(k) - \underset{\sim}{H}\hat{\underset{\sim}{R}}(k|k-1)] \tag{38}$$

and the vector of predicted values by

$$\hat{R}(k|k-1) = e^{-\underset{\sim}{b}T}\ \hat{\underset{\sim}{R}}(k-1|k-1) + e^{-\underset{\sim}{b}T}\ \underset{\sim}{U}(k-1), \tag{39}$$

where $\underset{\sim}{G}(k)$ is the vector of the Kalman filter gains.

If we are interested in the overall system reliability and if as in the missile case the subsystems are independent and in series, we have the output equation

$$R(k) = \prod_{i=1}^{4} R_i(k)$$

which can be added to the control signal flow graph as shown in Figure 8. For subsystems arranged in parallel or series-parallel, the system can be handled in a similar manner, and needed changes in the output equation will not influence the system itself.

In this chapter we applied tools from control theory to our reliability model and showed how to find the limits of the rework effectiveness α for which the system is stable and the final value of the item reliability when k becomes large. We also showed how a Kalman filter may be applied to smooth the estimated values of item reliability and feed back this estimated values upon which the decision about α is made.

In the next chapter we will show how the decision about α can be optimized with respect to a given performance measure.

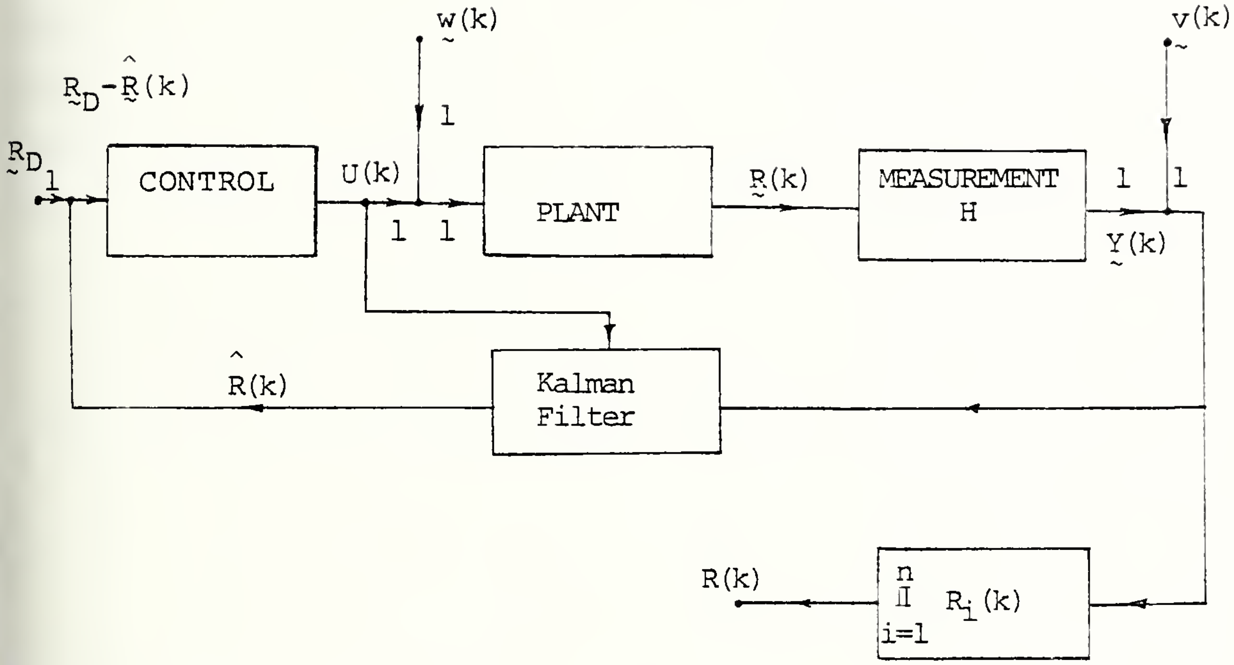

FIGURE 8. Signal flow graph for multidimensional rework control system

V. OPTIMIZATION

In the previous chapter we have studied our control model for the case we have linear equations and a single level of rework effectiveness α. In this chapter we will extend this study for non-linear equations and several levels of rework effectiveness, and show how to find the sequence of rework that optimizes a given performance criteria. First we will discuss various performance criteria that could be used for our control model, and then we will show how to find the optimal sequence of rework levels by dynamic programming.

A. THE PERFORMANCE MEASURE

The optimal control problem in our thesis is to find a control u(k) as defined by the expression

$$u(k) = \alpha(k) \, (R_D - R(k)) \, , \qquad (40)$$

which causes the system to follow a trajectory that optimizes a performance measure J. The rework effectiveness, $\alpha(k)$, is now allowed to change from step to step and in fact the problem now becomes one of finding the sequence of rework effectiveness $\alpha(k)$ that yields the optimal sequence of control u(k). NOtice that since $\alpha(k)$ is no longer constant, Expression (40) is non-linear.

First let us consider a trajectory (that is, the sequence of values the item reliability follows over time) that attains the desired reliability with a smallest number of reworks. This type of problem is called a "Minimum-Time" problem and the performance measure to be minimized may be generally stated as

$$J = \int_{t_o}^{t_f} dt = t_f - t_o$$

for the case of our discrete system, this becomes

$$J = \sum_{k=0}^{N-1} T ,$$

where (N-1) is the first period the desired reliability, R_D, is attained.

If instead we want to minimize the deviation of the first state of our system from its desired value we have the type of problem called a "Terminal Control" problem. Here, possible performance measures are

$$J = \sum_{i}^{N} R_D - R_{si}(k) ,$$

where the summation is done in all dimensions or reliabilities of the sub-systems. If positive and negative deviations

are equally undesirable these deviations should be squared.

To transfer a system from an arbitrary initial state R_o to some specified desired value with minimum expenditure of control effort or with minimum cost, we need to minimize,

$$J = \sum_{k=0}^{N-1} C(k)T = \sum_{k=0}^{N-1} cu(k)T ,$$

where we assume that the cost of a rework C(k) is proportional to the control u(k) or equivalently that the cost is proportional to the reliability improvement, $R_s(k) - R(k)$, after each rework.

Since these criteria are completely distinct in their concepts, we shall follow the usual approach of using a combination of them, that for the one dimensional problems takes the form,

$$J = HN[R_D - R_s(k)] + \sum_{k=0}^{N-1} [Q' + cu(k)]T$$

or

$$J = HN[R_D - R_s(k)] + \sum_{k=0}^{N-1} [Q + Gu(k)] , \qquad (41)$$

where HN, Q and G are relative weights so that by adjusting their values we can weight the relative importance of each criteria with respect to the other. These weighting factors can be functions of time if the relative importance varies with time. Notice that now the performance measure is a combination of criteria and has no physical meaning.

It should be noticed also that the rework effectiveness $\alpha(k)$ now has the definition given at the end of Chapter III:

$$\alpha(k) = \frac{R_s(k) - R(k)}{R_D - R(k)} ,$$

and that even when R_D is not equal to 1.0 we do not need the decision rule

$$M = \begin{cases} 0 & \text{if } R_D - R(k) < 0 \\ \alpha(k) & \text{if } R_D - R(k) > 0 \end{cases}$$

because the constraints in $\alpha(k)$ do not permit α to become negative.

B. OPTIMIZATION USING DYNAMIC PROGRAMMING

We now wish to show how the performance of our rework control system may be optimized in terms of the combined performance measure (41). An optimal solution would be expressed as a sequence of reworks that minimizes this criterion. We shall approach the problem with the method of dynamic programming, as developed by Bellman.

The basic notion here is given by Bellman's principle of optimality:

"An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision" [5]. Thus if $J^*_{k+1,N}(R(k+1))$ is the minimum cost to reach the final state at $k = N$, starting from the state $R(k+1)$ at time $t = (k+1)T$, and $J_{k,k+1}$ is the cost to move from kth to the (k+1)th state, then $C^*_{kN}(R(k),u(k))$, the minimum cost to go from the kth to the (k+1)th state when we use a particular control $u(k)$ is given by

$$C^*_{kN}(R(k),u(k)) = J_{k,k+1}(R(k),u(k)) + J^*_{k+1,N}(R(k+1)). \tag{42}$$

The optimal decision at the instant k, $u^*(k)$, is the decision that minimizes $C^*_{kN}(R(k),u(k))$ over the set of possible controls $u(k)$,

$$J^*_{kN}(R(k)) = \min_{u(k)} [C^*_{kN}(R(k),u(k))] . \tag{43}$$

Expressions (42) and (43) form the functional equations of our dynamic programming approach. The optimal sequence of decisions are built up from the final state N backwards toward the earlier states. This is necessary in order that J^* be known prior to the calculation of C^*. The values of $R(k)$ are given by our model, and the constraints of the

problem are

$$0.0 \leq R(k) \leq 1.0$$

and

$$0.0 \leq \alpha(k) \leq 1.0$$

The first step in the computational procedure, then, is to find the optimal policy for the last stage of operation. This is essentially a matter of trying all of the allowable control values at each of the allowable state values. To limit the required number of calculations, and make the computational procedure feasible, the allowable state and control values are discretized. The degree of approximation depends on the separation of these discrete values and on the method of interpolation used and can, of course, be adjusted.

For each discrete value of $R(N-1)$ we try all discrete values of $u(N-1)$ and calculate the resulting state $R(N)$. The optimal control for this rework is the one which yields the minimum cost. The procedure is repeated for all the other discrete values of $R(N-1)$. This gives a table of optimal policy for each value of $R(N-1)$ at the last stage. Since the cost $J_{N-1,N}$ is dependent on the value of the state $R(N-1)$ and on the value of the input applied, $u(N-1)$,

the minimum cost J* and the optimal control u*(N-1) are dependent on the value of the state R(N-1). For the last rework we use only the first term (terminal control) of our performance measure (41), but for the other, successive stages we must compute each term in that expression. When a state does not coincide with one of the discrete values we have to use interpolation to find the corresponding value of the performance measure. Since a direct search is used to solve the functional equations, the solution obtained is regarded as the global minimum.

A flow chart describing the computational procedure is shown in Figure 9. A Fortran program for this flow chart is described in Appendix B.

As an example of the use of this dynamic programming procedure, we sought the optimal sequence of reworks for the case where $b = 0.001$, $T = 100$, and constraints,

$$0.8 \leq R(k) \leq 1.0$$

and

$$0.7 \leq \alpha(k) \leq 1.0 .$$

We chose to work with twenty discrete values for $R(k)$, thirty discrete values for $\alpha(k)$, and four reworks. We input in the program the equations of our model:

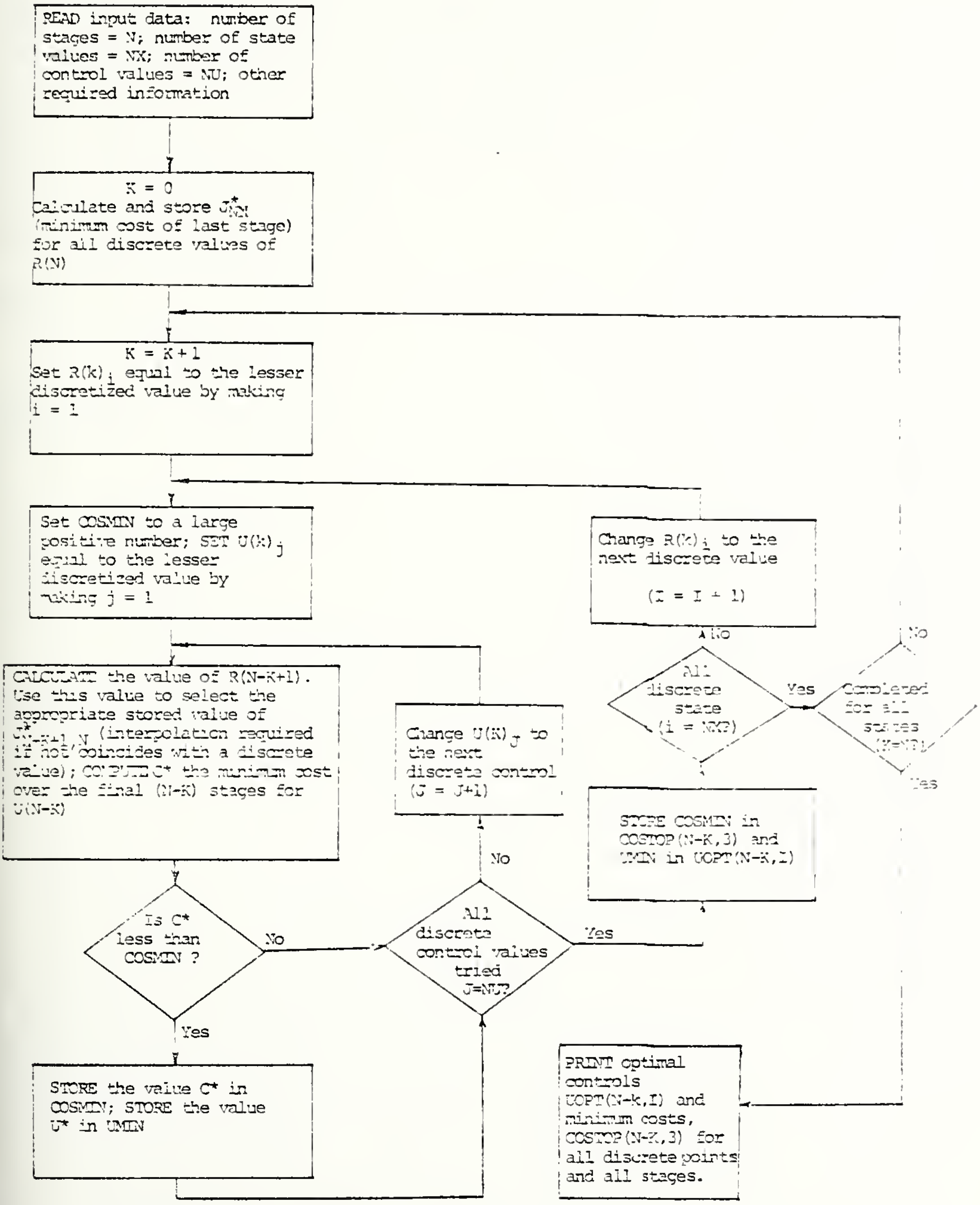
READ input data: number of stages = N; number of state values = NX; number of control values = NU; other required information
K = 0
Calculate and store J^*_{NX} (minimum cost of last stage) for all discrete values of R(N)
K = K + 1
Set $R(k)_i$ equal to the lesser discretized value by making i = 1
Set COSMIN to a large positive number; SET $U(k)_j$ equal to the lesser discretized value by making j = 1
CALCULATE the value of R(N-K+1). Use this value to select the appropriate stored value of $J^*_{N-K+1,N}$ (interpolation required if not coincides with a discrete value); COMPUTE C* the minimum cost over the final (N-K) stages for U(N-K)
Is C* less than COSMIN ?
No
Yes
STORE the value C* in COSMIN; STORE the value U* in UMIN
Change $U(K)_J$ to the next discrete control (J = J+1)
No
All discrete control values tried J=NU?
Yes
Change $R(k)_i$ to the next discrete value (I = I + 1)
No
All discrete state (i = NX?)
Yes
Completed for all states (K=N?)
No
Yes
STORE COSMIN in COSTOP(N-K,3) and UMIN in UOPT(N-K,I)
PRINT optimal controls UOPT(N-k,I) and minimum costs, COSTOP(N-K,3) for all discrete points and all stages.

FIGURE 9. Flow chart of the computational procedure for the dynamic programming solution for optimal control of the rework control system

$$R_s(k) = R(k) + \alpha(k)\ (1-R(k))$$

$$R(k+1) = R_s(k)\ e^{-bT} ,$$

and considered as input variable the rework effectiveness $\alpha(k)$ since in this program the equations need not be linear and this is the variable of immediate interest for the user. For the performance measure J given by Expression (41) we considered four distinct types of problems obtained by changing the weights. These are:

(1) Minimum Cost, so that the weights of the combined criterion J (41) are

$$G = 1.0 , \quad Q = 0.0 \quad \text{and} \quad HN = 0.0 ,$$

(2) Terminal State, or minimizing deviations from the final value. Here the weights in J are

$$G = 0.0 , \quad Q = 0.0 \quad \text{and} \quad HN = 1.0 ,$$

(3) Terminal State and Minimum Cost, where we assume a desired reliability $R_D = 0.99$, and a relatively important terminal state represented by the weights

$$G = 0.9 , \quad Q = 0.0 \quad \text{and} \quad HN = 500.0 ,$$

and

(4) The linear combination of the three criteria represented by the weights

$$G = 0.05 \;, \quad Q = 0.05 \quad \text{and} \quad HN = 500.0.$$

The dynamic programming solutions for the four cases are presented for each rework period in the Computer Output Section of this thesis. From these solutions we can now find, for each of the four cases, the optimal sequence of reworks for a given initial reliability R_o (entering with this value at the first rework table and, interpolating among the values listed, we can follow the process until the final state is reached).

The results are in Tables II to V for two values of initial reliability of an item. As we can see from these tables, the sequence of reworks that minimizes our performance measure depends on the value of the initial reliability R_o, for given values of the desired reliability R_D and of weightings in the performance measure.

Summarizing, we have applied a dynamic programming procedure to our control model of the reliability of an item and were able, for the four selected cases, to find an optimal sequence of reworks that minimizes a selected performance measure. For other selections of performance measure J, relative weightings HN, Q and G, constraints, initial reliability R_o and model parameters b and T, we

TABLE II

RELIABILITY FOLLOWING REWORK FOR INITIAL RELIABILITY R_o, AND OPTIMAL SEQUENCE OF REWORK LEVELS $\alpha(k)$

Case (1) - Minimum Cost

PERFORMANCE MEASURE: $J = [R_D - R_s(k)]H + \sum_{k=0}^{N-1} Q + GU(k)$

with $H = 0.0$, $Q = 0.0$, $G = 1.0$ and $U(k) = \alpha(k)$

$R_o = 0.8$

k	$R_s(k)$	$\alpha(k)$	$R_s(k+1)$
0	0.800	0.60	0.890
1	0.890	0.60	0.924
2	0.924	0.60	0.933
3	0.933	0.60	0.936
4	0.926		

$R_o = 0.9$

k	$R_s(k)$	$\alpha(k)$	$R_s(k+1)$
0	0.900	0.60	0.926
1	0.926	0.60	0.933
2	0.933	0.60	0.936
3	0.936	0.60	0.937
4	0.937		

TABLE III

RELIABILITY FOLLOWING REWORK FOR INITIAL RELIABILITY R_o, AND OPTIMAL SEQUENCE OF REWORK LEVELS $\alpha(k)$

Case (2) - Terminal State

PERFORMANCE MEASURE: $J = [R_D - R_s(k)]H + \sum_{k=0}^{N-1} Q+GU(k)$

with $R_D = 0.99$, $H = 1.0$, $Q = 0.0$, $G = 0.0$

and $U(k) = \alpha(k)$

$R_o = 0.8$

k	$R_s(k)$	$\alpha(k)$	$R_s(k+1)$
0	0.800	0.90	0.972
1	0.972	0.90	0.988
2	0.988	0.90	0.990
3	0.990	0.90	0.990
4	0.990		

$R_o = 0.9$

k	$R_s(k)$	$\alpha(k)$	$R_s(k+1)$
0	0.900	0.90	0.982
1	0.982	0.90	0.989
2	0.989	0.90	0.990
3	0.990	0.90	0.990
4	0.990		

TABLE IV

RELIABILITY FOLLOWING REWORK FOR INITIAL RELIABILITY R_o AND OPTIMAL SEQUENCE OF REWORK LEVELS $\alpha(k)$

Case (3) - Terminal State with Minimum Cost

PERFORMANCE MEASURE: $J = [R_D - R_s(k)]H + \sum_{k=0}^{N-1} Q + GU(k)$

with R_D 0.99 , $H = 500.0$, $Q = 0.0$, $G = 0.9$

and $U(k) = \alpha(k)$

$R_o = 0.8$

k	$R_s(k)$	$\alpha(k)$	$R_s(k+1)$
0	0.8	0.60	0.890
1	0.890	0.60	0.924
2	0.924	0.60	0.933
3	0.933	0.867	0.979
4	0.979		

$R_o = 0.9$

k	$R_s(k)$	$\alpha(k)$	$R_s(k+1)$
0	0.900	0.60	0.926
1	0.926	0.60	0.933
2	0.933	0.60	0.936
3	0.936	0.864	0.978
4	0.978		

TABLE V

RELIABILITY FOLLOWING REWORK FOR INITIAL RELIABILITY R_o, AND OPTIMAL SEQUENCE OF REWORK LEVELS $\alpha(k)$

Case (4) - Terminal State in Minimum Time with Minimal Cost

PERFORMANCE MEASURE: $J = [R_D - R_s(k)]H + \sum_{k=0}^{N-1} Q + GU(k)$,

with $R_D = 0.99$, $H = 500.0$, $Q = 0.05$, $G = 0.05$

and $U(k) = \alpha(k)$

$R_o = 0.8$

k	$R_s(k)$	$\alpha(k)$	$R_s(k+1)$
0	0.800	0.60	0.890
1	0.890	0.60	0.924
2	0.924	0.868	0.977
3	0.977	0.90	0.989
4	0.989		

$R_o = 0.9$

k	$R_s(k)$	$\alpha(k)$	$R_s(k+1)$
0	0.900	0.60	0.926
1	0.926	0.60	0.933
2	0.933	0.835	0.973
3	0.973	0.90	0.988
4	0.988		

would obtain different sequences of rework effectiveness $\alpha(k)$ that optimizes the selected performance measure. A suitable selection of the performance measure and relative weightings being a responsibility of the manager according to his purposes, experience and data available. Thus the cost of each rework, the minimum value allowable to the item reliability, minimum and maximum levels of rework that are physically feasible to implement, and other factors like the urgency to attain a desired reliability in minimum time, or a constraint in the budget, are taken into account when selecting the weightings in the combination of all the criteria.

In the next chapter we will present the general conclusions of this thesis and suggest the areas where one could do further study involving the rework control model.

VI. CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER STUDY

A simple, closed-loop inventory storage system from which the stored items are removed, sent through a rework mechanism to improve their reliability, and returned to storage has been investigated in this thesis. Such a system might be a stock of large ordnance which is acquired, stored and reworked at regular intervals.

A general rework model was developed by using the tools of control theory and a solution in closed form was found that permits further analysis of the system in a much easier way than if we had a solution in the form of series. This model also permitted us to find in a direct way the values of effectiveness of the rework that makes the system unstable and the final value of the item reliability over several reworks when the system is stable.

A Kalman filter was used in the control model to obtain an estimation of the item reliability when we have random error in our measure of the item reliability or when the rework process introduces randomness in the item reliability. This model was extended to more than one dimension for the case where we have systems composed of several subsystems in series, parallel, or both. Finally a study was made for the case where we can have several levels of rework and want to find the optimal sequence of such reworks that minimizes a performance criteria that leads to the

case of minimum time, minimum cost, terminal state or a linear combination of these criteria.

This thesis shows that control theory may be applied for administrative problems where a mathematical model describing the system can be derived. Control theory gives a broader understanding of the problem and due to its flexibility would permit us to include in the model many other variables not studied in this thesis.

Among the several interesting areas which might be pursued in further study, one is to determine methods for measuring or estimating the item reliability by frequent diagnostic checks so that this measurement process can be incorporated in the model. Another area is to broaden the field of the rework model so that other state and input variables could be incorporated. These include the probability that an item chosen at random might have reliability exceeding some predetermined reliability requirement, and other input variables (such as items expended for training purposes, use, or obsolescence, as well as new items acquired periodicaly). The rate of rework, the rate of expenditure, or the rate of acquisition could then be sought as control variables besides the rework effectiveness. Non-linear functions caused by the decision rule of doing a rework only when the item reliability is smaller than the desired value, can also be solved in terms of non-linear control theory.

A reliability control model has been developed in this thesis for a system of stored items requiring rework. It is hoped that the results presented here will not only be useful to inventory managers and high-level planners but will also generate further interest in the application of control theory to administrative problems.

APPENDIX A

Z TRANSFORM

A.1 DEFINITIONS AND PROPERTIES

The solution of difference equations by the z-transform method is very useful, because we can transform difference equations into algebraic equations in z. Once solved this algebraic equation we can then find the inverse z transform of this solution to obtain the solution in the time domain.

Thus, given a discrete function of t, f(kT), the z-transform of this function, symbolized by F(z), is defined by

$$Z[f(kT)] = F(z) \triangleq \sum_{k=0}^{\infty} f(kT)z^{-k} \tag{44}$$

From this definition we can obtain several useful properties. For example, let's find the z-transform of f((k+1)T):

$$Z[f(k+1)T)] = \sum_{k=0}^{\infty} f((k+1)T)z^{-k} = \sum_{k=1}^{\infty} f(kT)z^{-k+1}$$

$$= z[\sum_{k=0}^{\infty} f(kT)z^{-k} - f(0)] = z\ F(z) - zf(0). \tag{45}$$

Thus when a difference equation is transformed into an algebraic equation in z by the z-transform method, the initial data are automatically included in the algebraic representation.

Other useful properties are

$$z[af(kT)] = a\,F(z) \tag{46}$$

and

$$z[1(kT)] = \frac{z}{z-1}\,, \tag{47}$$

where 1(t) is the unit-step function:

$$1(kT) = \begin{cases} 0 & \text{for} \quad t < 0 \\ 1 & \text{for} \quad t > 0 \end{cases}$$

These two properties permit us to find the z transform of a constant, since any constant a can be expressed by the product a1(kT):

$$z[a] = az/z-1 \tag{48}$$

Another property is the distribution property

$$z[f_1(kT) + f_2(kT)] = z[f_1(kT)]+z[f_2(kT)] \tag{49}$$

The z-transform of the most common functions are given in several references listed in the Bibliography [1], [4] and [10].

The inverse transform of F(z) is f(kT), and is denoted by

$$Z^{-1}[F(z)] = f(kT) \quad . \tag{50}$$

From the tables of z-transform we can find the inverse of the simplest function. More complicated z-transforms may have to be expanded into partial fractions so that the tables can be used. Normally, we expand F(z)/z instead of F(z) into partial fractions because this leads to functions with z in the numerator after we multiply back by z, and the functions of z appearing in tables of z transforms usually have the factor z in their numerators.

When the partial fraction expansion does not give tabulated functions, we may have to find the appropriate z-transform from the definition. This will be the case in our problem.

Similar properties of z-transforms exist for inverse z transforms and will be useful:

$$Z^{-1}[F(z)/z] = Z^{-1}[z^{-1}F(z)] = f[(k-1)T] \ , \tag{51}$$

$$Z^{-1}[aF(z)] = aZ^{-1}[F(z)] = af(kT) \ , \tag{52}$$

$$Z^{-1}[\frac{az}{z-1}] = a\ 1(kT) = a \quad \text{for} \quad t > 0\ , \qquad (53)$$

and

$$Z^{-1}[F_1(z) + F_2(z)] = Z^{-1}[F_1(z) + Z^{-1}[F_2(z)]\ . \qquad (54)$$

The characteristic equation of a system is equal to the denominator of the overall transfer function and the stability of the system can be determined from the location of the roots of the characteristic equation [10]. A condition for stability is that all roots must lie inside the unit circle or

$$|z_i| < 1 \qquad (55)$$

Another important property of the z transform is the Final Value Theorem [10]:

"If x(t) has the z transform X(z) and X(z) has no poles (roots) outside the unit circle ($|Z| < 1$, that is the condition for stability), then the final value of x(t) or x(k) is given by

$$\lim_{t\to\infty} x(t) = \lim_{k\to\infty} x(k) = \lim_{z\to 1}[(z-1)X(z)]." \qquad (56)$$

To prove this note that

$$Z[x(k)] = X(z) = \sum_{k=0}^{\infty} x(k)\ z^{-k}$$

$$Z[x(k+1)] = ZX(z) - Zx(0) = \sum_{k=0}^{\infty} x(k+1)\ z^{-k}$$

Hence

$$zX(z) - zx(0) - X(z) = (z-1)X(z) - zx(0)$$

$$= \sum_{k=0}^{\infty} x(k+1)\ z^{-k} - \sum_{k=0}^{\infty} x(k)\ z^{-k}$$

from which we obtain

$$(z-1)X(z) = zx(0) + \sum_{k=0}^{\infty} [x(k+1) - x(k)]\ z^{-k}$$

Because of the assumed stability condition, we obtain, as $z \to 1$,

$$\lim_{z \to 1} [(z-1)X(z)] = x(0) + x(\infty) - x(0) = x(\infty)$$

which is Equation (56).

A.2. Z TRANSFORMS OF EQUATIONS (8) AND (9)

We found that the reliability immediately before the (k+1)th rework is given by equation (57).

$$R(k+1) = (1-\alpha)e^{-bT}R(k) + \alpha e^{-bT} \tag{57}$$

Since $(1-\alpha)e^{-bT}$ and $\alpha R_D e^{-bT}$ are constants, we can apply the properties (45), (46), (47), (48) and (49) to find the z-transform of both sides:

$$Z[R((k+1)T)] = Z[(1-\alpha)e^{-bT}R(kT)] + Z[\alpha e^{-bT}]$$

$$= Z[R((k+1)T)] = (1-\alpha)e^{-bT}Z[R(kT)] + \alpha e^{-bT}Z[1(kT)]$$

$$zR(z) - ZR(1) = (1-\alpha)e^{-bT}R(z) + \alpha e^{-bT}\frac{z}{z-1}, \qquad (58)$$

which is Equation (12) of Chapter I when we use R(1) as the initial condition.

Similarly for Equation (8):

$$R_s((k+1)T) = (1-\alpha)R_s(kT)e^{-bT} + \alpha \qquad (59)$$

$$zR_s(z) - zR(0) = (1-\alpha)e^{-bT}R_s(z) + \alpha\frac{z}{z-1} \qquad (60)$$

which is Equation (10) of Chapter I.

A.3. EXPANSION INTO PARTIAL FRACTIONS

Equations (58) and (60) can be solved for $R_s(z)$ and $R(z)$, giving:

$$R_s(z) = \frac{z}{z-(1-\alpha)e^{-bT}}R(0) + \frac{\alpha z}{[z-(1-\alpha)e^{-bT}][z-1]} \qquad (61)$$

and

$$R(z) = \frac{z}{z-(1-\alpha)e^{-bT}} R(1) + \frac{\alpha e^{-bT} z}{[z-(1-\alpha)e^{-bT}][z-1]} . \qquad (62)$$

In order to have simpler expressions we can proceed now to expand into partial fractions. We will exemplify it for the last equation only, since they differ only by a constant factor in the last term.

The first term has already a simple form so that we need only to expand the last term. For reasons stated in Section A.2, we will expand this term divided by z:

$$\frac{1}{z} \frac{\alpha e^{-bT} z}{[z-(1-\alpha)e^{-bT}][z-1]} = \frac{\alpha e^{-bT}}{[z-(1-\alpha)e^{-bT}][z-1]}$$

$$= \frac{A}{z-1} + \frac{B}{z-(1-\alpha)e^{-bT}} \qquad (63)$$

To determine the coefficient A we can multiply both sides by (z-1) and evaluate the expression at z = 1:

$$\left.\frac{\alpha e^{-bT}}{z-(1-\alpha)e^{-bT}}\right|_{z=1} = A + \left[\left.\frac{B(z-1)}{z-(1-\alpha)e^{-bT}}\right|_{z=1}\right.$$

This gives

$$A = \frac{\alpha e^{-bT}}{1-(1-\alpha)e^{-bT}} .$$

To evaluate B we multiply both sides by $[z-(1-\alpha)e^{-bT}]$ and evaluate at $z = (1-\alpha)e^{-bT}$, so that the term with A becomes zero:

$$\left|\frac{\alpha e^{-bT}}{z-1}\right|_{z=(1-\alpha)e^{-bT}} = 0 + B ,$$

this gives

$$B = \frac{\alpha e^{-bT}}{(1-\alpha)e^{-bT}-1} ,$$

Substituting back into (63), we have

$$\frac{\alpha e^{-bT}}{(z-1)[z-(1-\alpha)e^{-bT}]} = \frac{\alpha e^{-bT}}{[1-(1-\alpha)e^{-bT}][z-1]} + \frac{\alpha e^{-bT}}{[(1-\alpha)e^{-bT}-1][z-(1-\alpha)e^{-bT}]} ,$$

and R(z) can now be put in the form:

$$R(z) = \frac{z}{z-(1-\alpha)e^{-bT}}R(1) + \frac{z}{z-1}\,\frac{\alpha e^{-bT}}{(1-(1-\alpha)e^{-bT})} + \frac{z}{[z-(1-\alpha)e^{-bT}]}\,\frac{\alpha e^{-bT}}{[(1-\alpha)e^{-bT}-1]} \qquad (64)$$

The expansion for $R_s(z)$ follows immediately from this last equation:

$$R_s(z) = \frac{z}{z-(1-\alpha)e^{-bT}}R(0) + \frac{z}{(z-1)}\frac{\alpha}{[1-(1-\alpha)e^{-bT}]}$$

$$+ \frac{z}{[z-(1-\alpha)e^{-bT}]}\frac{\alpha}{[(1-\alpha)e^{-bT}-1]} \qquad (65)$$

A.4. INVERSE Z TRANSFORM

The first and last terms of Equations (64) and (65) are not found in the tables of z-transform so that we will proceed now to develop the expressions of these z-transforms from a guessed form of f(kT).

Since the first and last terms are identical in terms in z, differing only by constant factors, we need to do the development just for the expression in z.

Thus, let's assume that

$$f(kT) = (1-\alpha)^k e^{-bkT}$$

Applying the definition of z transform we have

$$F(z) = \sum_{k=0}^{\infty} f(kT)z^{-k} = \sum_{k=0}^{\infty} (1-\alpha)^k e^{-bkT} z^{-k} \qquad (66)$$

Multiplying both sides by $(1-\alpha)e^{-bT}z^{-1}$, we have:

$$(1-\alpha)e^{-bT}z^{-1}F(z) = \sum_{k=0}^{\infty} (1-\alpha)^{k+1} e^{-b(k+1)T} z^{-(k+1)} \qquad (67)$$

Subtracting (67) from (66) side by side we get

$$[1-(1-\alpha)e^{-bT}z^{-1}]F(z) = \sum_{k=0}^{\infty}(1-\alpha)^{k}e^{-bkT}z^{-k} - \sum_{k=0}^{\infty}(1-\alpha)^{k+1}e^{-b(k+1)T}z^{-(k+1)}$$

or

$$[1-(1-\alpha)e^{-bT}z^{-1}]F(z) = 1 + \sum_{k=0}^{\infty}(1-\alpha)^{k+1}e^{-b(k+1)T}z^{-(1+1)}$$

$$- \sum_{k=0}^{\infty}(1-\alpha)^{k+1}e^{-b(k+1)T}z^{-(k+1)}$$

or

$$[1-(1-\alpha)e^{-bT}z^{-1}]F(z) = 1$$

Thus

$$F(z) = \frac{1}{1-(1-\alpha)e^{-bT}z^{-1}} = \frac{z}{z-(1-\alpha)e^{-bT}} \qquad (68)$$

From this it follows that

$$z^{-1}\left[\frac{z}{z-(1-\alpha)e^{-bT}}\right] = f(kT) = (1-\alpha)^{k}e^{-bkT} \qquad (69)$$

Also,

$$z^{-1}\left[\frac{z}{z-(1-\alpha)e^{-bT}} \quad \frac{\alpha}{(1-\alpha)e^{-bT}-1}\right] = \frac{\alpha(1-\alpha)^{k}e^{-bkT}}{(1-\alpha)e^{-bT}-1} \qquad (70)$$

and we can get similar expressions by changing the constant part. From (52), (54), (69) and (70) we can now find the inverse z-transform of equations (64) and (65):

$$R(kT) = (1-\alpha)^k e^{-bkT} R(1) + \frac{\alpha e^{-bT}}{1-(1-\alpha)e^{-bT}} + \frac{\alpha(1-\alpha)^k e^{-b(k+1)T}}{(1-\alpha)e^{-bT}-1}$$

or

$$R(kT) = (1-\alpha)^k e^{-bkT} R(1) + \frac{\alpha e^{-bT}}{(1-\alpha)e^{-bT}-1}[(1-\alpha)^k e^{-bkT}-1] \tag{71}$$

And from (65)

$$R_s(kT) = (1-\alpha)^k e^{-bkT} R_o + \frac{\alpha}{(1-\alpha)e^{-bT}-1}[(1-\alpha)^k e^{-bkT}-1] \tag{72}$$

These are the equations (13) and (11) of Chapter I, respectively, and constitute the solution of the difference equations (57) and (59) in the time domain.

APPENDIX B

COMPUTER PROGRAM DISCUSSION

In Chapter V we have shown that the functional equations of the dynamic programming to compute the optimal costs over the set of possible controls are

$$C^*_{kN}(R(k),u(k)) = J_{k,k+1}(R(k),u(k))+J^*_{k+1,N}(R(k+1))$$

and

$$J^*_{kN}(R(k)) = \min_{u(k)} [C^*_{kN}(R(k),u(k))]$$

The program that performs these computations is listed in the Computer Program Section of this thesis and is composed of three subroutines and two functions. The subroutine QUANTU computes the discrete values of the state variable R(k), and of the inputs U(k). For each discrete value, the subroutine STATE computes the value of the state variable at the next state and the dynamic equations of our model (15) and (16) has to be input in this subroutine. With the vlaue of R(k+1), the subroutine cost selects the value of the performance measure J by interpolation between the two nearest discrete values stored in the array of discrete values of optimal cost, COSTOP, of the (k+1)th state. This is possible because the computation is done

backwards from the last stage of operation. The costs of the last stage are computed by the Function HN, and the subsequent cost for each state and each value of the input U are computed by the Function G. These costs are summed by the subroutine COST and in the main program the minimum value is found for all discrete values of U(k). These computations are repeated for all the discrete values of the state variable as can be seen in the flow chart of figure 9.

The program output presents the optimal costs and corresponding effectiveness of rework α to go to the final state for all discrete values of the reliability so that starting with a given initial reliability at the initial state or first rework table we can, interpolating through the values of the next state until the final state, find out the optimal sequence of rework for that particular initial condition.

The program output for the examples of Chapter V are presented in the Computer Output Section of this thesis.

```
        COMPUTER OUTPUT    CASE 1

       DATA INPUT

       N     NX      NU  MODE
       4     20      30     0

     XMIN  XMAX  UMIN  UMAX
      0.3   1.0   0.5   0.9




COSTS OF OPERATION AND OPTIMAL CONTROL LAW

     TABLE FOR THE FINAL STATE
     -------------------------
        FINAL       MINIMUM
        STATE        COST
     -------------------------
        1.00         0.0
        0.99         0.0
        0.98         0.0
        0.97         0.0
        0.96         0.0
        0.95         0.0
        0.94         0.0
        0.93         0.0
        0.92         0.0
        0.91         0.0
        0.89         0.0
        0.88         0.0
        0.87         0.0
        0.86         0.0
        0.85         0.0
        0.84         0.0
        0.83         0.0
        0.82         0.0
        0.81         0.0
        0.80         0.0
     -------------------------
```

COMPUTER OUTPUT   CASE 1

```
            TABLE FOR THE 4TH REWORK
----------------------------------------------------------
 CURRENT       NEXT        MINIMUM       OPTIMAL
  RS(K)      RS(K+1)        COST         REWORK
----------------------------------------------------------

  1.00        0.962         0.4           0.60
----------------------------------------------------------
  0.99        0.958         0.4           0.60
----------------------------------------------------------
  0.98        0.954         0.4           0.60
----------------------------------------------------------
  0.97        0.951         0.4           0.60
----------------------------------------------------------
  0.96        0.947         0.4           0.60
----------------------------------------------------------
  0.95        0.943         0.4           0.60
----------------------------------------------------------
  0.94        0.939         0.4           0.60
----------------------------------------------------------
  0.93        0.935         0.4           0.60
----------------------------------------------------------
  0.92        0.931         0.4           0.60
----------------------------------------------------------
  0.91        0.928         0.4           0.60
----------------------------------------------------------
  0.89        0.924         0.4           0.60
----------------------------------------------------------
  0.88        0.920         0.4           0.60
----------------------------------------------------------
  0.87        0.916         0.4           0.60
----------------------------------------------------------
  0.86        0.912         0.4           0.60
----------------------------------------------------------
  0.85        0.909         0.4           0.60
----------------------------------------------------------
  0.84        0.905         0.4           0.60
----------------------------------------------------------
  0.83        0.901         0.4           0.60
----------------------------------------------------------
  0.82        0.897         0.4           0.60
----------------------------------------------------------
  0.81        0.893         0.4           0.60
----------------------------------------------------------
  0.80        0.890         0.4           0.60
----------------------------------------------------------
```

COMPUTER OUTPUT   CASE 1

```
              TABLE FOR THE 8TH REWORK
------------------------------------------------------------
 CURRENT       NEXT         MINIMUM        OPTIMAL
  RS(K)      RS(K+1)         COST          REWORK
------------------------------------------------------------

   1.00       0.962          0.7            0.60
------------------------------------------------------------

   0.99       0.958          0.7            0.60
------------------------------------------------------------

   0.98       0.954          0.7            0.60
------------------------------------------------------------

   0.97       0.951          0.7            0.60
------------------------------------------------------------

   0.96       0.947          0.7            0.60
------------------------------------------------------------

   0.95       0.943          0.7            0.60
------------------------------------------------------------

   0.94       0.939          0.7            0.60
------------------------------------------------------------

   0.93       0.935          0.7            0.60
------------------------------------------------------------

   0.92       0.931          0.7            0.60
------------------------------------------------------------

   0.91       0.928          0.7            0.60
------------------------------------------------------------

   0.89       0.924          0.7            0.60
------------------------------------------------------------

   0.88       0.920          0.7            0.60
------------------------------------------------------------

   0.87       0.916          0.7            0.60
------------------------------------------------------------

   0.86       0.912          0.7            0.60
------------------------------------------------------------

   0.85       0.909          0.7            0.60
------------------------------------------------------------

   0.84       0.905          0.7            0.60
------------------------------------------------------------

   0.83       0.901          0.7            0.60
------------------------------------------------------------

   0.82       0.897          0.7            0.60
------------------------------------------------------------

   0.81       0.893          0.7            0.60
------------------------------------------------------------

   0.80       0.890          0.7            0.60
------------------------------------------------------------
```

COMPUTER OUTPUT CASE 1

```
            TABLE FOR THE 2TH REWORK
------------------------------------------------------------
 CURRENT        NEXT         MINIMUM         OPTIMAL
  RS(K)       RS(K+1)         COST           REWORK
------------------------------------------------------------

   1.00        0.962          1.1             0.60
------------------------------------------------------------
   0.99        0.958          1.1             0.60
------------------------------------------------------------
   0.98        0.954          1.1             0.60
------------------------------------------------------------
   0.97        0.951          1.1             0.60
------------------------------------------------------------
   0.96        0.947          1.1             0.60
------------------------------------------------------------
   0.95        0.943          1.1             0.60
------------------------------------------------------------
   0.94        0.939          1.1             0.60
------------------------------------------------------------
   0.93        0.935          1.1             0.60
------------------------------------------------------------
   0.92        0.931          1.1             0.60
------------------------------------------------------------
   0.91        0.928          1.1             0.60
------------------------------------------------------------
   0.89        0.924          1.1             0.60
------------------------------------------------------------
   0.88        0.920          1.1             0.60
------------------------------------------------------------
   0.87        0.916          1.1             0.60
------------------------------------------------------------
   0.86        0.912          1.1             0.60
------------------------------------------------------------
   0.85        0.909          1.1             0.60
------------------------------------------------------------
   0.84        0.905          1.1             0.60
------------------------------------------------------------
   0.83        0.901          1.1             0.60
------------------------------------------------------------
   0.82        0.897          1.1             0.60
------------------------------------------------------------
   0.81        0.893          1.1             0.60
------------------------------------------------------------
   0.80        0.890          1.1             0.60
------------------------------------------------------------
```

COMPUTER OUTPUT   CASE 1

```
           TABLE FOR THE 1TH REWORK
-------------------------------------------------
 CURRENT      NEXT        MINIMUM      OPTIMAL
 RS(K)      RS(K+1)       COST         REWORK
-------------------------------------------------

  1.00       0.962         1.4          0.60
-------------------------------------------------
  0.99       0.958         1.4          0.60
-------------------------------------------------
  0.98       0.954         1.4          0.60
-------------------------------------------------
  0.97       0.951         1.4          0.60
-------------------------------------------------
  0.96       0.947         1.4          0.60
-------------------------------------------------
  0.95       0.943         1.4          0.60
-------------------------------------------------
  0.94       0.939         1.4          0.60
-------------------------------------------------
  0.93       0.935         1.4          0.60
-------------------------------------------------
  0.92       0.931         1.4          0.60
-------------------------------------------------
  0.91       0.928         1.4          0.60
-------------------------------------------------
  0.89       0.924         1.4          0.60
-------------------------------------------------
  0.88       0.920         1.4          0.60
-------------------------------------------------
  0.87       0.916         1.4          0.60
-------------------------------------------------
  0.86       0.912         1.4          0.60
-------------------------------------------------
  0.85       0.909         1.4          0.60
-------------------------------------------------
  0.84       0.905         1.4          0.60
-------------------------------------------------
  0.83       0.901         1.4          0.60
-------------------------------------------------
  0.82       0.897         1.4          0.60
-------------------------------------------------
  0.81       0.893         1.4          0.60
-------------------------------------------------
  0.80       0.890         1.4          0.60
-------------------------------------------------
```

```
                    COMPUTER OUTPUT    CASE 2


                   DATA  INPUT

                   N     NX     NU  MODE
                   4     20     30     0

                XMIN XMAX UMIN UMAX
                 0.8  1.0  0.6  0.9




         COSTS OF OPERATION AND OPTIMAL CONTROL LAW

              TABLE FOR THE FINAL STATE
              -------------------------
                 FINAL        MINIMUM
                 STATE         COST
              -------------------------
                  1.00         0.00
                  0.99         0.00
                  0.98         0.00
                  0.97         0.00
                  0.96         0.00
                  0.95         0.00
                  0.94         0.00
                  0.93         0.00
                  0.92         0.01
                  0.91         0.01
                  0.89         0.01
                  0.88         0.01
                  0.87         0.01
                  0.86         0.02
                  0.85         0.02
                  0.84         0.02
                  0.83         0.03
                  0.82         0.03
                  0.81         0.03
                  0.80         0.04
              -------------------------
```

```
                    COMPUTER OUTPUT    CASE 2


          TABLE FOR THE 4TH REWORK
---------------------------------------------------------------
 CURRENT       NEXT         MINIMUM         OPTIMAL
  RS(K)      RS(K+1)         COST           REWORK
---------------------------------------------------------------

   1.00        0.989          0.0             0.89
---------------------------------------------------------------

   0.99        0.990          0.0             0.90
---------------------------------------------------------------

   0.98        0.989          0.0             0.90
---------------------------------------------------------------

   0.97        0.988          0.0             0.90
---------------------------------------------------------------

   0.96        0.987          0.0             0.90
---------------------------------------------------------------

   0.95        0.986          0.0             0.90
---------------------------------------------------------------

   0.94        0.985          0.0             0.90
---------------------------------------------------------------

   0.93        0.984          0.0             0.90
---------------------------------------------------------------

   0.92        0.983          0.0             0.90
---------------------------------------------------------------

   0.91        0.982          0.0             0.90
---------------------------------------------------------------

   0.89        0.981          0.0             0.90
---------------------------------------------------------------

   0.88        0.980          0.0             0.90
---------------------------------------------------------------

   0.87        0.979          0.0             0.90
---------------------------------------------------------------

   0.86        0.978          0.0             0.90
---------------------------------------------------------------

   0.85        0.977          0.0             0.90
---------------------------------------------------------------

   0.84        0.976          0.0             0.90
---------------------------------------------------------------

   0.83        0.975          0.0             0.90
---------------------------------------------------------------

   0.82        0.974          0.0             0.90
---------------------------------------------------------------

   0.81        0.973          0.0             0.90
---------------------------------------------------------------

   0.80        0.972          0.0             0.90
---------------------------------------------------------------
```

COMPUTER OUTPUT CASE 2

```
                TABLE FOR THE 3TH REWORK
------------------------------------------------------------
 CURRENT        NEXT            MINIMUM          OPTIMAL
  RS(K)       RS(K+1)            COST            REWORK
------------------------------------------------------------
   1.00        0.990             0.0              0.90
------------------------------------------------------------
   0.99        0.990             0.0              0.90
------------------------------------------------------------
   0.98        0.989             0.0              0.90
------------------------------------------------------------
   0.97        0.988             0.0              0.90
------------------------------------------------------------
   0.96        0.987             0.0              0.90
------------------------------------------------------------
   0.95        0.986             0.0              0.90
------------------------------------------------------------
   0.94        0.985             0.0              0.90
------------------------------------------------------------
   0.93        0.984             0.0              0.90
------------------------------------------------------------
   0.92        0.983             0.0              0.90
------------------------------------------------------------
   0.91        0.982             0.0              0.90
------------------------------------------------------------
   0.89        0.981             0.0              0.90
------------------------------------------------------------
   0.88        0.980             0.0              0.90
------------------------------------------------------------
   0.87        0.979             0.0              0.90
------------------------------------------------------------
   0.86        0.978             0.0              0.90
------------------------------------------------------------
   0.85        0.977             0.0              0.90
------------------------------------------------------------
   0.84        0.976             0.0              0.90
------------------------------------------------------------
   0.83        0.975             0.0              0.90
------------------------------------------------------------
   0.82        0.974             0.0              0.90
------------------------------------------------------------
   0.81        0.973             0.0              0.90
------------------------------------------------------------
   0.80        0.972             0.0              0.90
------------------------------------------------------------
```

COMPUTER OUTPUT   CASE 2

```
        TABLE FOR THE 2TH REWORK
------------------------------------------------------------
 CURRENT        NEXT        MINIMUM         OPTIMAL
  RS(K)       RS(K+1)        COST           REWORK
------------------------------------------------------------

   1.00        0.990          0.0             0.90
------------------------------------------------------------

   0.99        0.990          0.0             0.90
------------------------------------------------------------

   0.98        0.989          0.0             0.90
------------------------------------------------------------

   0.97        0.988          0.0             0.90
------------------------------------------------------------

   0.96        0.987          0.0             0.90
------------------------------------------------------------

   0.95        0.986          0.0             0.90
------------------------------------------------------------

   0.94        0.985          0.0             0.90
------------------------------------------------------------

   0.93        0.984          0.0             0.90
------------------------------------------------------------

   0.92        0.983          0.0             0.90
------------------------------------------------------------

   0.91        0.982          0.0             0.90
------------------------------------------------------------

   0.89        0.981          0.0             0.90
------------------------------------------------------------

   0.88        0.980          0.0             0.90
------------------------------------------------------------

   0.87        0.979          0.0             0.90
------------------------------------------------------------

   0.86        0.978          0.0             0.90
------------------------------------------------------------

   0.85        0.977          0.0             0.90
------------------------------------------------------------

   0.84        0.976          0.0             0.90
------------------------------------------------------------

   0.83        0.975          0.0             0.90
------------------------------------------------------------

   0.82        0.974          0.0             0.90
------------------------------------------------------------

   0.81        0.973          0.0             0.90
------------------------------------------------------------

   0.80        0.972          0.0             0.90
------------------------------------------------------------
```

COMPUTER OUTPUT CASE 2

```
              TABLE FOR THE 1TH REWORK
---------------------------------------------------------
 CURRENT        NEXT        MINIMUM         OPTIMAL
  RS(K)       RS(K+1)        COST           REWORK
---------------------------------------------------------

   1.00        0.990          0.0             0.90
---------------------------------------------------------

   0.99        0.990          0.0             0.90
---------------------------------------------------------

   0.98        0.989          0.0             0.90
---------------------------------------------------------

   0.97        0.988          0.0             0.90
---------------------------------------------------------

   0.96        0.987          0.0             0.90
---------------------------------------------------------

   0.95        0.986          0.0             0.90
---------------------------------------------------------

   0.94        0.985          0.0             0.90
---------------------------------------------------------

   0.93        0.984          0.0             0.90
---------------------------------------------------------

   0.92        0.983          0.0             0.90
---------------------------------------------------------

   0.91        0.982          0.0             0.90
---------------------------------------------------------

   0.89        0.981          0.0             0.90
---------------------------------------------------------

   0.88        0.980          0.0             0.90
---------------------------------------------------------

   0.87        0.979          0.0             0.90
---------------------------------------------------------

   0.86        0.978          0.0             0.90
---------------------------------------------------------

   0.85        0.977          0.0             0.90
---------------------------------------------------------

   0.84        0.976          0.0             0.90
---------------------------------------------------------

   0.83        0.975          0.0             0.90
---------------------------------------------------------

   0.82        0.974          0.0             0.90
---------------------------------------------------------

   0.81        0.973          0.0             0.90
---------------------------------------------------------

   0.80        0.972          0.0             0.90
---------------------------------------------------------
```

```
                    COMPUTER OUTPUT    CASE 3

                  DATA INPUT

                  N    NX     NU  MODE
                  4    20     30     0

                XMIN  XMAX  UMIN  UMAX
                 0.8   1.0   0.6   0.9



          COSTS OF OPERATION AND OPTIMAL CONTROL LAW

               TABLE FOR THE FINAL STATE
               -------------------------
                  FINAL        MINIMUM
                  STATE         COST
               -------------------------
                  1.00          0.05
                  0.99          0.00
                  0.98          0.06
                  0.97          0.25
                  0.96          0.52
                  0.95          0.91
                  0.94          1.41
                  0.93          2.03
                  0.92          2.75
                  0.91          3.59
                  0.89          4.54
                  0.88          5.60
                  0.87          6.76
                  0.86          8.04
                  0.85          9.44
                  0.84         10.94
                  0.83         12.55
                  0.82         14.27
                  0.81         16.11
                  0.80         18.05
               -------------------------
```

COMPUTER OUTPUT CASE 3

```
           TABLE FOR THE 4TH REWORK
-------------------------------------------------------------
 CURRENT        NEXT         MINIMUM          OPTIMAL
  RS(K)      RS(K+1)          COST            REWORK
-------------------------------------------------------------

   1.00        0.979           0.6              0.78
-------------------------------------------------------------

   0.99        0.979           0.6              0.80
-------------------------------------------------------------

   0.98        0.979           0.7              0.82
-------------------------------------------------------------

   0.97        0.979           0.7              0.83
-------------------------------------------------------------

   0.96        0.978           0.7              0.84
-------------------------------------------------------------

   0.95        0.978           0.7              0.85
-------------------------------------------------------------

   0.94        0.978           0.7              0.86
-------------------------------------------------------------

   0.93        0.979           0.7              0.87
-------------------------------------------------------------

   0.92        0.979           0.8              0.88
-------------------------------------------------------------

   0.91        0.980           0.8              0.89
-------------------------------------------------------------

   0.89        0.979           0.8              0.89
-------------------------------------------------------------

   0.88        0.980           0.8              0.90
-------------------------------------------------------------

   0.87        0.979           0.8              0.90
-------------------------------------------------------------

   0.86        0.978           0.8              0.90
-------------------------------------------------------------

   0.85        0.977           0.8              0.90
-------------------------------------------------------------

   0.84        0.976           0.8              0.90
-------------------------------------------------------------

   0.83        0.975           0.9              0.90
-------------------------------------------------------------

   0.82        0.974           0.9              0.90
-------------------------------------------------------------

   0.81        0.973           0.9              0.90
-------------------------------------------------------------

   0.80        0.972           0.9              0.90
-------------------------------------------------------------
```

COMPUTER OUTPUT CASE 3

```
                TABLE FOR THE 3TH REWORK
----------------------------------------------------------------
 CURRENT       NEXT         MINIMUM          OPTIMAL
 RS(K)       RS(K+1)         COST            REWORK
----------------------------------------------------------------

   1.00       0.962           1.0             0.60
----------------------------------------------------------------
   0.99       0.958           1.0             0.60
----------------------------------------------------------------
   0.98       0.954           1.0             0.60
----------------------------------------------------------------
   0.97       0.951           1.0             0.60
----------------------------------------------------------------
   0.96       0.947           1.0             0.60
----------------------------------------------------------------
   0.95       0.943           1.0             0.60
----------------------------------------------------------------
   0.94       0.939           1.1             0.60
----------------------------------------------------------------
   0.93       0.935           1.1             0.60
----------------------------------------------------------------
   0.92       0.931           1.1             0.60
----------------------------------------------------------------
   0.91       0.928           1.1             0.60
----------------------------------------------------------------
   0.89       0.924           1.1             0.60
----------------------------------------------------------------
   0.88       0.920           1.1             0.60
----------------------------------------------------------------
   0.87       0.916           1.1             0.60
----------------------------------------------------------------
   0.86       0.912           1.1             0.60
----------------------------------------------------------------
   0.85       0.909           1.1             0.60
----------------------------------------------------------------
   0.84       0.905           1.1             0.60
----------------------------------------------------------------
   0.83       0.901           1.1             0.60
----------------------------------------------------------------
   0.82       0.897           1.1             0.60
----------------------------------------------------------------
   0.81       0.893           1.1             0.60
----------------------------------------------------------------
   0.80       0.890           1.1             0.60
----------------------------------------------------------------
```

COMPUTER OUTPUT   CASE 3

```
            TABLE FOR THE 2TH REWORK
----------------------------------------------------------
 CURRENT       NEXT         MINIMUM         OPTIMAL
  RS(K)      RS(K+1)         COST           REWORK
----------------------------------------------------------

   1.00       0.962          1.4             0.60
----------------------------------------------------------

   0.99       0.958          1.4             0.60
----------------------------------------------------------

   0.98       0.954          1.4             0.60
----------------------------------------------------------

   0.97       0.951          1.4             0.60
----------------------------------------------------------

   0.96       0.947          1.4             0.60
----------------------------------------------------------

   0.95       0.943          1.4             0.60
----------------------------------------------------------

   0.94       0.939          1.4             0.60
----------------------------------------------------------

   0.93       0.935          1.4             0.60
----------------------------------------------------------

   0.92       0.931          1.4             0.60
----------------------------------------------------------

   0.91       0.928          1.4             0.60
----------------------------------------------------------

   0.89       0.924          1.4             0.60
----------------------------------------------------------

   0.88       0.920          1.4             0.60
----------------------------------------------------------

   0.87       0.916          1.4             0.60
----------------------------------------------------------

   0.86       0.912          1.4             0.60
----------------------------------------------------------

   0.85       0.909          1.4             0.60
----------------------------------------------------------

   0.84       0.905          1.4             0.60
----------------------------------------------------------

   0.83       0.901          1.4             0.60
----------------------------------------------------------

   0.82       0.897          1.4             0.60
----------------------------------------------------------

   0.81       0.893          1.4             0.60
----------------------------------------------------------

   0.80       0.890          1.4             0.60
----------------------------------------------------------
```

COMPUTER OUTPUT CASE 2

```
           TABLE FOR THE 1TH REWORK
---------------------------------------------------------
 CURRENT       NEXT         MINIMUM        OPTIMAL
  RS(K)      RS(K+1)         COST          REWORK
---------------------------------------------------------

   1.00        0.962          1.7           0.60
---------------------------------------------------------
   0.99        0.958          1.7           0.60
---------------------------------------------------------
   0.98        0.954          1.7           0.60
---------------------------------------------------------
   0.97        0.951          1.7           0.60
---------------------------------------------------------
   0.96        0.947          1.7           0.60
---------------------------------------------------------
   0.95        0.943          1.7           0.60
---------------------------------------------------------
   0.94        0.939          1.7           0.60
---------------------------------------------------------
   0.93        0.935          1.7           0.60
---------------------------------------------------------
   0.92        0.931          1.7           0.60
---------------------------------------------------------
   0.91        0.928          1.7           0.60
---------------------------------------------------------
   0.89        0.924          1.7           0.60
---------------------------------------------------------
   0.88        0.920          1.7           0.60
---------------------------------------------------------
   0.87        0.916          1.7           0.60
---------------------------------------------------------
   0.86        0.912          1.7           0.60
---------------------------------------------------------
   0.85        0.909          1.7           0.60
---------------------------------------------------------
   0.84        0.905          1.7           0.60
---------------------------------------------------------
   0.83        0.901          1.7           0.60
---------------------------------------------------------
   0.82        0.897          1.7           0.60
---------------------------------------------------------
   0.81        0.893          1.7           0.60
---------------------------------------------------------
   0.80        0.890          1.7           0.60
```

```
          COMPUTER OUTPUT    CASE 4

         DATA INPUT

         N     NX     NU  MODE
         4     20     30     0

      XMIN  XMAX  UMIN  UMAX
       0.8   1.0   0.6   0.9



C0STS OF OPERATION AND OPTIMAL CONTROL LAW

     TABLE FOR THE FINAL STATE
     -------------------------
        FINAL        MINIMUM
        STATE         COST
     -------------------------
         1.00          0.05
         0.99          0.00
         0.98          0.06
         0.97          0.25
         0.96          0.52
         0.95          0.91
         0.94          1.42
         0.93          2.03
         0.92          2.75
         0.91          3.59
         0.89          4.54
         0.88          5.60
         0.87          6.76
         0.86          8.04
         0.85          9.44
         0.84         10.94
         0.83         12.55
         0.82         14.27
         0.81         16.11
         0.80         18.05
     -------------------------
```

```
                COMPUTER OUTPUT    CASE 4

         TABLE FOR THE 4TH REWORK
-------------------------------------------------------
 CURRENT      NEXT        MINIMUM       OPTIMAL
 RS(K)      RS(K+1)        COST         REWORK
-------------------------------------------------------

  1.00       0.989          0.1           0.89
-------------------------------------------------------

  0.99       0.990          0.1           0.90
-------------------------------------------------------

  0.98       0.989          0.1           0.90
-------------------------------------------------------

  0.97       0.988          0.1           0.90
-------------------------------------------------------

  0.96       0.987          0.1           0.90
-------------------------------------------------------

  0.95       0.986          0.1           0.90
-------------------------------------------------------

  0.94       0.985          0.1           0.90
-------------------------------------------------------

  0.93       0.984          0.1           0.90
-------------------------------------------------------

  0.92       0.983          0.1           0.90
-------------------------------------------------------

  0.91       0.982          0.1           0.90
-------------------------------------------------------

  0.89       0.981          0.1           0.90
-------------------------------------------------------

  0.88       0.980          0.1           0.90
-------------------------------------------------------

  0.87       0.979          0.2           0.90
-------------------------------------------------------

  0.86       0.978          0.2           0.90
-------------------------------------------------------

  0.85       0.977          0.2           0.90
-------------------------------------------------------

  0.84       0.976          0.2           0.90
-------------------------------------------------------

  0.83       0.975          0.2           0.90
-------------------------------------------------------

  0.82       0.974          0.2           0.90
-------------------------------------------------------

  0.81       0.973          0.2           0.90
-------------------------------------------------------

  0.80       0.972          0.3           0.90
-------------------------------------------------------
```

COMPUTER OUTPUT CASE 4

```
              TABLE FOR THE 8TH REWORK
---------------------------------------------------------------
 CURRENT        NEXT         MINIMUM          OPTIMAL
  RS(K)       RS(K+1)         COST            REWORK
---------------------------------------------------------------

   1.00        0.962          0.2              0.60
---------------------------------------------------------------

   0.99        0.953          0.2              0.60
---------------------------------------------------------------

   0.98        0.954          0.2              0.60
---------------------------------------------------------------

   0.97        0.957          0.2              0.65
---------------------------------------------------------------

   0.96        0.959          0.2              0.69
---------------------------------------------------------------

   0.95        0.964          0.2              0.74
---------------------------------------------------------------

   0.94        0.969          0.2              0.80
---------------------------------------------------------------

   0.93        0.975          0.2              0.85
---------------------------------------------------------------

   0.92        0.979          0.2              0.88
---------------------------------------------------------------

   0.91        0.980          0.2              0.89
---------------------------------------------------------------

   0.89        0.978          0.2              0.89
---------------------------------------------------------------

   0.88        0.980          0.2              0.90
---------------------------------------------------------------

   0.87        0.979          0.2              0.90
---------------------------------------------------------------

   0.86        0.978          0.2              0.90
---------------------------------------------------------------

   0.85        0.977          0.2              0.90
---------------------------------------------------------------

   0.84        0.976          0.2              0.90
---------------------------------------------------------------

   0.83        0.975          0.2              0.90
---------------------------------------------------------------

   0.82        0.974          0.2              0.90
---------------------------------------------------------------

   0.81        0.973          0.2              0.90
---------------------------------------------------------------

   0.80        0.972          0.2              0.90
---------------------------------------------------------------
```

COMPUTER OUTPUT   CASE 4

```
            TABLE FOR THE 2TH REWORK
----------------------------------------------------
CURRENT      NEXT        MINIMUM       OPTIMAL
 RS(K)     RS(K+1)        COST         REWORKK
----------------------------------------------------

  1.00      0.962          0.2           0.60
----------------------------------------------------

  0.99      0.958          0.2           0.60
----------------------------------------------------

  0.98      0.954          0.2           0.60
----------------------------------------------------

  0.97      0.951          0.2           0.60
----------------------------------------------------

  0.96      0.947          0.2           0.60
----------------------------------------------------

  0.95      0.943          0.3           0.60
----------------------------------------------------

  0.94      0.939          0.3           0.60
----------------------------------------------------

  0.93      0.935          0.3           0.60
----------------------------------------------------

  0.92      0.931          0.3           0.60
----------------------------------------------------

  0.91      0.928          0.3           0.60
----------------------------------------------------

  0.89      0.924          0.3           0.60
----------------------------------------------------

  0.88      0.920          0.3           0.60
----------------------------------------------------

  0.87      0.916          0.3           0.60
----------------------------------------------------

  0.86      0.912          0.3           0.60
----------------------------------------------------

  0.85      0.909          0.3           0.60
----------------------------------------------------

  0.84      0.905          0.3           0.60
----------------------------------------------------

  0.83      0.901          0.3           0.60
----------------------------------------------------

  0.82      0.897          0.3           0.60
----------------------------------------------------

  0.81      0.893          0.3           0.60
----------------------------------------------------

  0.80      0.890          0.3           0.60
----------------------------------------------------
```

COMPUTER OUTPUT CASE 4

```
           TABLE FOR THE 1TH REWORK
---------------------------------------------------------
 CURRENT       NEXT        MINIMUM          OPTIMAL
  RS(K)      RS(K+1)        COST            REWORK
---------------------------------------------------------

   1.00       0.962          0.3              0.60
---------------------------------------------------------
   0.99       0.958          0.3              0.60
---------------------------------------------------------
   0.98       0.954          0.3              0.60
---------------------------------------------------------
   0.97       0.951          0.3              0.60
---------------------------------------------------------
   0.96       0.947          0.3              0.60
---------------------------------------------------------
   0.95       0.943          0.3              0.60
---------------------------------------------------------
   0.94       0.939          0.3              0.60
---------------------------------------------------------
   0.93       0.935          0.3              0.60
---------------------------------------------------------
   0.92       0.931          0.3              0.60
---------------------------------------------------------
   0.91       0.928          0.3              0.60
---------------------------------------------------------
   0.89       0.924          0.3              0.60
---------------------------------------------------------
   0.88       0.920          0.3              0.60
---------------------------------------------------------
   0.87       0.916          0.3              0.60
---------------------------------------------------------
   0.86       0.912          0.3              0.60
---------------------------------------------------------
   0.85       0.909          0.3              0.60
---------------------------------------------------------
   0.84       0.905          0.3              0.60
---------------------------------------------------------
   0.83       0.901          0.3              0.60
---------------------------------------------------------
   0.82       0.897          0.3              0.60
---------------------------------------------------------
   0.81       0.893          0.3              0.60
---------------------------------------------------------
   0.80       0.890          0.3              0.60
---------------------------------------------------------
```

COMPUTER PROGRAM

```
C
C.....................................................................
C
C                              DYNCON
C
C     IDENTIFICATION
C
C          TITLE      -DYNCON(OPTIMAL CONTROL USING DYNAMIC PROGRAMMING)
C
C          PROGRAMMER -FERREIRA P. A.
C
C          THESIS RESEARCH
C
C          THESIS ADVISOR -GLENN FRANK LINDSAY PH. D.
C
C          DATE       -DEC. 1978
C
C
C     PURPOSE
C
C       TO FIND THE OPTIMAL CONTROL TRAJECTORY AND THE ASSOCIATED MINIMUM
C       COST FOR FIRST ORDER DIFFERENTIAL EQUATION SYSTEMS APPROXIMATED
C       BY A DIFFERENCE EQUATION AND PERFORMANCE MEASURE APPROXIMATED BY
C       A SUMMATION . THE ALLOWABLE STATE AND CONTROL VALUES ARE QUANTI-
C       ZED TO MAKE THE COMPUTATION FEASIBLE.
C
C
C     DESCRIPTION OF PARAMETERS
C
C         XMIN   LOWER BOUND OF THE STATE VARIABLE (CONSTRAINT).
C         XMAX   UPPER BOUND OF THE STATE VARIABLE (CONSTRAINT).
C         UMIN   LOWER BOUND OF THE CONTROL VARIABLE (CONSTRAINT).
C         UMAX   UPPER BOUND OF THE CONTROL VARIABLE (CONSTRAINT).
C       COSTOP   AN ARRAY (TWO DIMENSIONS) WITH THE OPTIMAL COST
C                FOR EACH QUANTIZED LEVEL AT EACH CURRENT STATE
C                VALUE.
C         UOPT   AN ARRAY (TWO DIMENSIONS) WITH THE OPTIMAL CONTROL
C                CORRESPONDING TO EACH QUANTIZED LEVEL AT EACH CU-
C                RRENT STATE VALUE .
C            N   NUMBER OF DISCRETE CURRENT STATE VALUES INCLUDING
C                THE INITIAL AND FINAL STATES .
C           NX   NUMBER OF QUANTIZED LEVELS FOR EACH CURRENT STATE
C                VALUE .
C           NU   NUMBER OF QUANTIZED LEVELS FOR THE CONTROL VARIABLE
C            X   CURRENT DISCRET VALUE OF THE STATE VARIABLE .
C            U   CURRENT QUANTIZED LEVEL OF THE CONTROL VARIABLE .
C            K   INDEX INDICATING THE CURRENT ORDERED STATE OF THE
C                STATE VARIABLE . AT THE INITIAL POINT K=1 .
```

```
C         I  INDEX INDICATING THE QUANTIZED LEVEL OF THE STATE           DYN00490
C            VARIABLE (VARIES FROM 1 TO NX) .                            DYN00500
C         J  INDEX INDICATING THE QUANTIZED LEVEL OF THE CONTROL         DYN00510
C            VARIABLE (VARIES FROM 1 TO NU) .                            DYN00520
C        KK  INDEX INDICATING THE CURRENT DISCRETE TIME . AT THE         DYN00530
C            INITIAL STATE KK=0.                                         DYN00540
C     IMODE  OPTION FOR TABLE PRINTOUT : IMODE=0 PRINTS ONLY THE         DYN00550
C            OPTIMAL CONTROL AND OPTIMAL COST AT EACH LEVEL ; IF         DYN00560
C            IMODE=1 PRINTS ALL QUANTIZED VALUES AT EACH LEVEL .         DYN00570
C                                                                        DYN00580
C                                                                        DYN00590
C    DATA CARDS                                                          DYN00600
C                                                                        DYN00610
C      FIRST:  NUMBER OF STATES (N<=6) , NUMBER OF STATE QUANTIZED       DYN00620
C              LEVELS (NX<=101) , NUMBER OF CONTROL VARIABLE QUAN-       DYN00630
C              TIZED LEVELS (NU<=101) , AND MODE OF TABLE PRINTOUT       DYN00640
C              IN COLUMNS 1-5,6-10,11-15,16-20 (RIGHT JUSTIFIED) .       DYN00650
C     SECOND:  MINIMUM AND MAXIMUM VALUES FOR THE CONSTRAINT OF THE      DYN00660
C              STATE VARIABLE AND MINIMUM AND MAXIMUM VALUE FOR THE      DYN00670
C              CONTROL VARIABLES (FORMAT 5.1) ,IM COLUMNS 1-5,6-10,      DYN00680
C              11-15,16-20 .                                             DYN00690
C      THIRD:  IDENTIFICATION LABEL IN COLUMNS 1-72 .                    DYN00700
C                                                                        DYN00710
C                                                                        DYN00720
C    SUBROUTINES AND FUNCTIONS REQUIRED                                  DYN00730
C                                                                        DYN00740
C        DYNCON REQUIRES TWO FUNCTIONS , HN(X) AND G(U) , AND ONE        DYN00750
C        SUBROUTINE WHERE THE NECESSARY EQUATIONS HAS TO BE INPUT        DYN00760
C        BY THE USER ACCORDING TO THE INSTRUCTIONS ALREADY EXISTING      DYN00770
C        IN EACH SUCH SUBPROGRAM .                                       DYN00780
C        DYNCON SUPPLIES THE  SUBROUTINE QUANTU THAT MAKES THE           DYN00790
C        DISCRETIZATION OF U AND X AND THE SUBROUTINE COST THAT          DYN00800
C        MAKES THE INTERPOLATION AND COMPUTE THE SUM OF COSTS FOR        DYN00810
C        THE SEVERAL STAGES .                                            DYN00820
C                                                                        DYN00830
C                                                                        DYN00840
C                                                                        DYN00850
C.......................................................................DYN00860
C                                                                        DYN00870
C                                                                        DYN00880
C                                                                        DYN00890
      COMMON/CONSTR/XMIN,XMAX,UMIN,UMAX,B,T                              DYN00900
      COMMON/PERFOR/COSTOP(6,101),UOPT(6,101)                            DYN00910
      REAL*8 ITITLE(9)                                                   DYN00920
C                                                                        DYN00930
C                                                                        DYN00940
1     FORMAT(9A8)                                                        DYN00950
3     FORMAT(1H1,/////,6X,9A8,///)                                       DYN00960
```

```
4      FORMAT(2F10.3)                                                   DYN00965
10     FORMAT(4I5)                                                      DYN00970
20     FORMAT(4F5.1)                                                    DYN00980
25     FORMAT(35X,'DATA INPUT',//,35X,'N',3X,'NX',3X,'NU',1X,'MODE',    DYN00990
      1/,31X,4I5,//,32X,'XMIN',1X,'XMAX',1X,'UMIN',1X,'UMAX')           DYN01000
26     FORMAT(31X,4F5.1,//////)                                         DYN01010
30     FORMAT(1H0,25X,'COSTS OF OPERATION AND OPTIMAL CONTROL LAW')     DYN01020
35     FORMAT(30X,'TABLE FOR THE ',I1,'TH REWORK',/,                    DYN01030
      121X,45('-'),/,22X,'CURRENT',5X,'NEXT',5X,'MINIMUM',5X,           DYN01040
      2'OPTIMAL',/,23X,'RS(K)',4X,'RS(K+1)',5X,'COST',7X,'REWORK ')     DYN01050
40     FORMAT(22X,F6.2,5X,F6.3,5X,F6.1,6X,F6.2)                         DYN01060
50     FORMAT(21X,45('-'),/)                                            DYN01070
51     FORMAT(31X,25('-'))                                              DYN01080
55     FORMAT(1H-,30X,'TABLE FOR THE FINAL STATE',/,31X,25('-'),/,      DYN01090
      134X,'FINAL',5X,'MINIMUM',/,34X,'STATE',6X,'COST',/,31X,25('-'))  DYN01100
60     FORMAT(33X,F6.2,5X,F6.2)                                         DYN01110
C                                                                       DYN01120
C           DATA INPUT                                                  DYN01130
C                                                                       DYN01140
       DATA BIG/1.0 E 70/                                               DYN01150
       READ(5,4) B,T                                                    DYN01155
       READ(5,10) N,NX,NU,IMODE                                         DYN01160
       READ(5,20) XMIN,XMAX,UMIN,UMAX                                   DYN01170
       READ(5,1) (ITITLE(I),I=1,9)                                      DYN01180
       WRITE(6,3) (ITITLE(I),I=1,9)                                     DYN01190
       WRITE(6,25) N,NX,NU,IMODE                                        DYN01200
       WRITE(6,26) XMIN,XMAX,UMIN,UMAX                                  DYN01210
       WRITE(6,30)                                                      DYN01220
       IFLAG=0                                                          DYN01230
C                                                                       DYN01240
C        INITIALIZE COST OF THE LAST STAGE                              DYN01250
C                                                                       DYN01260
       K=0                                                              DYN01270
       U=0.0                                                            DYN01280
       WRITE(6,55)                                                      DYN01290
       DO 80 JJ=1,NX                                                    DYN01300
          II=NX+1-JJ                                                    DYN01310
          CALL QUANTU(XMIN,XMAX,N,NX,JJ,X,QUANTX)                       DYN01320
          C=HN(X)                                                       DYN01330
          COSTOP(N+1,II)=C                                              DYN01340
          WRITE(6,60)X,C                                                DYN01350
80     CONTINUE                                                         DYN01360
       WRITE(6,51)                                                      DYN01370
C                                                                       DYN01380
C           MAIN LOOP                                                   DYN01390
C                                                                       DYN01400
       KK=N+1                                                           DYN01410
       DO 300 L=1,N                                                     DYN01420
```

```
        K=KK-L                                                          DYN01430
        WRITE(6,3) (ITITLE(I),I=1,9)                                    DYN01440
        WRITE(6,35) K                                                   DYN01450
        DO 200 I=1,NX                                                   DYN01460
            II=NX+1-I                                                   DYN01470
            IF(IFLAG.EQ.0)WRITE(6,50)                                   DYN01480
            CALL QUANTU(XMIN,XMAX,N,NX,I,X,QUANTX)                      DYN01490
            COSMIN=BIG                                                  DYN01500
            DO 100 J=1,NU                                               DYN01510
                CALL QUANTU(UMIN,UMAX,N,NU,J,U,QUANT)                   DYN01520
                CALL STATE(X,U,XP1,K,N,IFLAG)                           DYN01530
                IF(IFLAG.EQ.1)GO TO 90                                  DYN01540
                CALL COST(K,I,N,NX,XMIN,G(U),XP1,QUANTX,C)              DYN01550
                IF(IMODE.EQ.1)WRITE(6,40)X,XP1,C,U                      DYN01560
                IF(C.GT.COSMIN)GO TO 100                                DYN01570
                COSTOP(K,II)=C                                          DYN01580
                COSMIN=C                                                DYN01590
                UOPT(K,II)=U                                            DYN01600
                XP1OPT=XP1                                              DYN01610
90              IFLAG=0                                                 DYN01620
100         CONTINUE                                                    DYN01630
        IF(IMODE.EQ.0)WRITE(6,40)X,XP1OPT,COSTOP(K,II),UOPT(K,II)       DYN01640
200     CONTINUE                                                        DYN01650
        WRITE(6,50)                                                     DYN01660
300   CONTINUE                                                          DYN01670
      STOP                                                              DYN01680
      END                                                               DYN01690
C                                                                       DYN01700
C.......................................................................DYN01710
      SUBROUTINE COST(K,I,N,NX,XMIN,G,XP1,QUANTX,C)                     DYN01720
C.......................................................................DYN01730
C                                                                       DYN01740
      COMMON/PERFOR/COSTOP(6,101),UOPT(6,101)                           DYN01750
C                                                                       DYN01760
      DATA ZERTOL/1.0 E-05/                                             DYN01770
C         SELECT THE APPROPIATE STORED VALUE OF J                       DYN01780
      DEL=XP1-XMIN                                                      DYN01790
      DELIN=DEL/QUANTX                                                  DYN01800
      INDEL=INT(DELIN)+1                                                DYN01810
      FRAC=(DEL-FLOAT(INDEL-1)*QUANTX)/QUANTX                           DYN01820
      IF(FRAC.LT.ZERTOL.OR.(1.0-FRAC).LT.ZERTOL) GO TO 5                DYN01830
C         INTERPOLATION IF XP1 IS NOT A GRID POINT                      DYN01840
      CI=(COSTOP(K+1,INDEL+1)-COSTOP(K+1,INDEL))*FRAC+COSTOP(K+1,INDEL) DYN01850
      GO TO 10                                                          DYN01860
5     CI=COSTOP(K+1,INDEL)                                              DYN01870
10    C=CI+G                                                            DYN01880
      RETURN                                                            DYN01890
      END                                                               DYN01900
```

```
C.......................................................................DYN01910
      SUBROUTINE STATE(X,U,XP1,K,N,IFLAG)                               DYN01920
C.......................................................................DYN01930
C                                                                       DYN01940
      COMMON/CONSTR/XMIN,XMAX,UMIN,UMAX,B,T                             DYN01950
C                                                                       DYN01960
C                                                                       DYN01970
C         THE STATE DIFFERENCE EQUATIONS HAVE TO BE                     DYN01980
C         INPUT HERE . EXAMPLE: XP1=(1+A*DELT)*X + B*DELT*U             DYN01990
C                                                                       DYN02000
      XP=X*EXP(-B*T)                                                    DYN02010
      XP1=(1-U)*XP+U                                                    DYN02020
      IF(XP1.GT.XMAX.OR.XP1.LT.XMIN) IFLAG=1                            DYN02030
      IF(U.GT.UMAX.OR.U.LT.UMIN) IFLAG=1                                DYN02040
      RETURN                                                            DYN02050
      END                                                               DYN02060
C                                                                       DYN02070
C.......................................................................DYN02080
      FUNCTION HN(X)                                                    DYN02090
C.......................................................................DYN021CC
C                                                                       DYN02110
C                                                                       DYN02120
C         THE FUNCTION HN HAS TO BE INPUT HERE .                        DYN02130
C         EXAMPLE:  HN=X**2                                             DYN02140
C         IF THE FINAL VALUE IS FREE SET HN=0 .                         DYN02150
C                                                                       DYN02160
      HN=(0.99-X)*5000.0                                                DYN02170
      RETURN                                                            DYN02180
      END                                                               DYN02190
C                                                                       DYN02200
C.......................................................................DYN02210
      FUNCTION G(U)                                                     DYN02220
C.......................................................................DYN02230
C                                                                       DYN02240
C                                                                       DYN02250
C         THE FUNCTION G HAS TO BE INPUT HERE .                         DYN02260
C         EXAMPLE: G=U**2                                               DYN02270
C                                                                       DYN02280
      G=3.0*U+0.15                                                      DYN02290
      RETURN                                                            DYN02300
      END                                                               DYN02310
C                                                                       DYN02320
C.......................................................................DYN02330
      SUBROUTINE QUANTU(SMIN,SMAX,N,NS,M,S,QUANT)                       DYN02340
C.......................................................................DYN02350
C                                                                       DYN02360
      RM=M                                                              DYN02370
      RNS=NS                                                            DYN02380
```

```
      QUANT=(SMAX-SMIN)/(RNS-1.0)
      S=SMAX-(RM-1.0)*QUANT
      RETURN
      END
```

BIBLIOGRAPHY

1. Albert B. Bishop (1975), Introduction to Discrete Linear Control — Theory and Application. Academic Press, New York.

2. James Guy Bohannan, A Reliability Model for Stored Items Requiring Rework. M.S. Thesis. U.S. Naval Postgraduate School, Monterey, California. 1977.

3. Donald E. Kirk (1970), Optimal Control Theory — An Introduction. Prentice-Hall, Inc. Englewood Cliffs, New Jersey.

4. Benjamin C. Kuo (1977), Digital Control Systems. SRL Publishing Company, Champaign, Illinois, 61820.

5. Richard E. Bellman and Stuart E. Dreyfus, Applied Dynamic Programming (1962). Princeton University Press, Princeton, New Jersey.

6. Joog, R.V., and Craig, A.T., Introduction to Mathematical Statistics, The MacMillan Company, 1959.

7. Feller, W., An Introduction to Probability and Its Applications, John Wiley & Sons, 1957.

8. Glenn F. Lindsay, Reliability in Stored Ordnance: Selection of Items for Training Expenditure, Naval Postgraduate School report to OP-964.

9. Donald E. Kirk, Stochastic Control, Notes Naval Postgraduate School.

10. Ogata, K., Modern Control Engineering, Prentice-Hall, Inc. Englewood Cliffs, N.J.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Associate Professor Glenn F. Lindsay, Code 55Ls Department of Operations Research Naval Postgraduate School Monterey, California 93940	2
4.	LT. Ellen Franz Roland, Code 55Ro Department of Operations Research Naval Postgraduate School Monterey, California 93940	2
5.	CDR Paulo Antonio Ferreira aos cuidados de Antonio J. Ferreira R. Dr. Octario Kelly 20. apto. 403 Tijuca Rio de Janeiro 20000 BRASIL, SOUTH AMERICA	2

Thesis 179844
F2647 Ferreira
c.1 Reliability control
model for stored items
requiring rework.

thesF2647
Reliability control model for stored ite
3 2768 001 01500 1
DUDLEY KNOX LIBRARY

.1.